Theses and Dissertations   1. Thesis and Dissertation Collection, all items

2005-12

Reachable sets for multiple asteroid sample return missions

Croley, Patrick A.

Monterey, California. Naval Postgraduate School

http://hdl.handle.net/10945/1826

NAVAL POSTGRADUATE SCHOOL

MONTEREY, CALIFORNIA

THESIS

REACHABLE SETS FOR MULTIPLE ASTEROID SAMPLE RETURN MISSIONS

by

Patrick A. Croley

December 2005

Thesis Advisor: I. Michael Ross
Second Reader: Stacy Weinstein

Approved for public release; distribution is unlimited

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE		*Form Approved OMB No. 0704-0188*

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instruction, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188) Washington DC 20503.

1. AGENCY USE ONLY (*Leave blank*)	2. REPORT DATE December 2005	3. REPORT TYPE AND DATES COVERED Engineer's Thesis

4. TITLE AND SUBTITLE: Reachable Sets for Multiple Asteroid Sample Return Missions	5. FUNDING NUMBERS
6. AUTHOR(S) Croley, Patrick A.	

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey, CA 93943-5000	8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING /MONITORING AGENCY NAME(S) AND ADDRESS(ES) National Aeronautics and Space Administration Jet Propulsion Laboratory 4800 Oak Grove Drive Pasadena, CA 91109-8099	10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

12a. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release; distribution is unlimited	12b. DISTRIBUTION CODE

13. ABSTRACT (maximum 200 words)
In support of future NASA asteroid sample return missions, this thesis examines strategies to reduce the number of feasible asteroid targets. Reachable sets are defined in a reduced classical orbital element space. The boundary of this reduced space is obtained by extremizing a family of convex combinations of orbital elements. The resulting group of optimization problems is solved using a direct collocation pseudospectral technique by a MATLAB application package called DIDO. The reachable sets are examined to narrow the possible valid asteroid choices in order to aid in mission design and analysis of alternative targets. A solar electric propulsion system is modeled with the stay times at each asteroid, Earth departure, and Earth arrival hyperbolic excess velocities implemented as constrained optimization parameters.

14. SUBJECT TERMS Low Thrust, Optimization, Multiple Asteroid Mission			15. NUMBER OF PAGES 145
			16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified	20. LIMITATION OF ABSTRACT UL

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)

i

THIS PAGE INTENTIONALLY LEFT BLANK

REACHABLE SETS FOR MULTIPLE ASTEROID SAMPLE RETURN MISSIONS

Patrick A. Croley
Lieutenant Commander, U.S. Navy
B.S., Tulane University, 1996

Submitted in partial fulfillment of the
requirements for the degree of

ASTRONAUTICAL ENGINEER
and
MASTER OF SCIENCE IN ASTRONAUTICAL ENGINEERING

from the

NAVAL POSTGRADUATE SCHOOL
December 2005

Author: Patrick A. Croley

Approved by: I. Michael Ross
                   Thesis Advisor

                   Stacy Weinstein
                   Second Reader

                   Anthony J. Healy
                   Chairman, Department of Mechanical and Astronautical Engineering

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

In support of future NASA asteroid sample return missions, this thesis examines strategies to reduce the number of feasible asteroid targets. Reachable sets are defined in a reduced classical orbital element space. The boundary of this reduced space is obtained by extremizing a family of convex combinations of orbital elements. The resulting group of optimization problems is solved using a direct collocation pseudospectral technique by a MATLAB application package called DIDO. The reachable sets are examined to narrow the possible valid asteroid choices in order to aid in mission design and analysis of alternative targets. A solar electric propulsion system is modeled with optimal stay times at each asteroid, Earth departure, and Earth arrival hyperbolic excess velocities implemented as constrained optimization parameters. For choosing rendezvous and return mission candidate asteroids, the use of the outer approximation limits the feasible target quickly by an order of magnitude in a given mission.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

LIST OF FIGURES

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF TABLES

THIS PAGE INTENTIONALLY LEFT BLANK

ACKNOWLEDGMENTS

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF SYMBOLS AND ACRONYMS

Subscripts:

$0, i, j, f$ - Initial, intermediate or final conditions respectively

e, s, a - Earth, Sun, or Asteroid respectively

min, max, wet, dry - Minimum value, Maximum Value, Wet mass, Dry mass

Symbols:

a - Semi-major axis

a_{trans} - Semi-major axis for Hohmann transfer orbit

C_3 - V-infinity squared

ct - Thrust coefficients for engine model

cm - Mass flow rate coefficients for engine model

E - Endpoint or Mayer Cost Functional

$\bar{E}$ - Augmented Mayer Cost Functional

e - Eccentricity

$\mathbf{e}$ - Events function or state error vector

$\mathbf{e}_{L2}$ - Euclidean root mean square error

F - Lagrange Cost Functional

$\mathbf{f}$ - Right-hand side of Dynamic equations

ga - Solar array distance model coefficients

$\mathbf{h}$ - Path function

H - Hamiltonian or Absolute Magnitude (asteroid ephemeris)

$\bar{H}$ - Augmented Hamiltonian

i - Orbital inclination

I_{sp} - Specific Impulse

g_0 - Gravity constant at Earth Surface

J - Scalar cost functional (a.k.a. performance function or objective function)

L - Lagrangian or augmented function

m - Mass of spacecraft

m_{prop} - Mass of propellant on spacecraft

$\dot{m}$ - Mass flow rate

N   - Number of nodes

n - Mean motion of an orbit

P - Power available from the solar array

P_0 - Reference power at 1 AU and at beginning of life

P_e - Power of spacecraft available to propulsion units

P_h - Housekeeping power requirements

p   - Vector of parameters

p - Semi-latus rectum

p_i - i[th] polynomial coefficient of launch vehicle performance

R_e - Radial distance from the sun to Earth

S - Switching function

T   - Thrust vector

T - Magnitude of **T**

t, τ - Time

$tcons$ - Solar array time degradation model coefficients

U_{dist} — Non-dimensional unit term for distance

U — Arbitrary bounding set for controls

u — Control vector

u* — Optimal Control vector

V_∞ — Hyperbolic excess speed

V — Velocity Vector

V_{trans_a} — Velocity at first burn in Hohmann transfer

V_{trans_b} — Velocity at second burn in Hohmann transfer

v_r — Radial velocity component

v_t — Transverse or along-track velocity component

v_e — Exhaust velocity of engine

x — State vector for spacecraft

x^0 — Initial starting conditions for a state

$x^{\mathbf{f}}$ — Final conditions for a state

α — Weighting factor in convex combination cost function

β — Weighting factor in convex combination cost function

β_0 — Angle between the thrust vector and the local horizontal direction

γ — Weighting factor in convex combination cost function

ΔV — "Delta-V" or velocity imparted onto a spacecraft

λ_i — Lagrange multipliers for dynamic constraints

σ_i — Lagrange multipliers for event constraints

μ_i — Lagrange multipliers for path constraints

μ_s - Gravitational parameter of sun

ν - Spacecraft angular displacement

θ - Spacecraft thrust direction from transverse axis

ω - Argument of Periapsis

$\mathbf{\Psi}_i$ - optimization code solution state vector at node i

$\hat{\mathbf{\Psi}}_i$ - propagated solution state vector at node i

$\mathbf{\Psi}_{max}$ - maximum value of optimized solution for each state

Acronyms:

CMT	Covector Mapping Theorem
DOF	Degree of Freedom
JPL	Jet Propulsion Laboratory (Pasadena)
KKT	Karush-Kuhn-Tucker
LGL	Legrendre-Gauss-Labatto
NLP	Nonlinear Programming problem
NPS	Naval Postgraduate School
NSTAR	NASA SEP Technology Applications Readiness
ODE	Ordinary Differential Equation (solver)
PMP	Pontryagin's Minimum Principle
S/C	Spacecraft
SEP	Solar-Electric Propulsion
SEPTOP	Solar-Electric Propulsion Trajectory Optimization Programs
SOI	Sphere of Influence
TPBVP	Two Point Boundary Value Problem

I. INTRODUCTION

A. MULTIPLE ASTEROID SAMPLE RETURN MISSIONS

Since the last Apollo mission in the 1970's, only NASA's Genesis mission has returned extraterrestrial samples to Earth for analysis. Recently, several missions have been planned or launched that intend to return material from planets, asteroids, and comet tails with the availability of more efficient propulsion systems. Largely due to the success of Deep Space 1, low thrust ion and Hall engines are now feasible technologies for interplanetary missions. However, with the increase efficiency and performance of these thrusters comes increasingly difficult mission planning objectives. Unlike ballistic trajectories, continuous thrust capable missions are much more complex to plan and generally do not have closed form solutions. However, even though these missions have been studied for well over 40 years [Ref. 1], the additional complexities of mission planning for multiple rendezvous targets with a return to Earth provides ample subject for research.

Finding an optimal low thrust trajectory between Earth and an asteroid can be formulated as a two-point boundary value problem (TPBVP) [Ref. 2]. Finding an optimal trajectory to return a sample to Earth is also a TPBVP, however to solve both trajectories simultaneously is more difficult, but can be done with application of indirect or direct optimization methods. Indeed, if one can simultaneously solve for larger series of optimal trajectories, then some basic mission planning for a multiple asteroid sample returns can be completed. However, in the real world, the problem is rarely given as just finding an optimal trajectory to reach several pre-selected asteroid targets and return the samples to Earth. Since there are over 3000 asteroids with a perihelion distance less than 1.5 AU from which to choose, the problem is to find which asteroids can be reached with a given fuel loading. The number of feasible asteroids is maximized if fuel optimal trajectories can be found to reach them and return. Anything is possible given enough money, but real world spacecraft missions have a budget limit and thus a fuel limit.

Identifying all reachable targets for a given mission is important to conduct analysis of alternative missions and contingency planning in case a launch window is

1

missed the original targets are no longer desirable. This also reduces the number of feasible targets to select for possible missions based on size, type, or other scientific value. Finding the "reachable set" of possible asteroid targets for a two sample return mission is the focus of this thesis.

B. FUEL LIMITED OPTIMAL TRAJECTORIES

The primary goal in nearly every aspect of interplanetary spacecraft mission design is to minimize the fuel required to achieve the mission objectives. This, in turn provides for the lowest mission cost and highest scientific payload capacity. Although, minimizing the overall mission time can be a significant manpower cost savings, this is usually a secondary concern. Since many endeavors are limited by budgets, unlike the Apollo program, this monetary limit can be directly linked to available launch vehicles, spacecraft mass and thus available fuel. Usually through an iterative concept or mission design process general spacecraft parameters can be determined before any optimization is completed in the actual mission planning. Thus, for this thesis NASA's Jet Propulsion Laboratory (JPL) has provided a given set of mission parameters.

The trajectories displayed or discussed herein will be trajectories to extremize (maximize and minimize) the domain of possible rendezvous orbits. These rendezvous orbits will be characterized by the semi-major axis (a), eccentricity (e), and inclination (i) in heliocentric inertial coordinates. This reduced set of classical orbital elements describes a manifold in space that represents the basis of initially calculating the amount of fuel required to transfer from one manifold to another. This disregard for the actual position of a target asteroid at a given time, or phasing, ensures the mission can be formulated as a continuous optimization problem and to not introduce discrete integers that would require hybrid optimization methods. Thus, any asteroid located inside a reachable set will be a feasible target at some time in the orbit, but not for all time. Any phasing problems and specific optimal trajectories must be computed after selecting available targets inside of the reachable sets.

C. OTHER SOLUTION METHODOLOGIES

Of course there are other ways to reduce the domain of possible target sets in order to conduct mission planning. The first way is by brute force of computational power. For example, a search of JPL's Database of Asteroids and Comets (DASCOM) in

October of 2003 found 3072 asteroids with a perihelion distance less than 1.5 AU. Given a two asteroid sample return mission, each of the almost 9.5 million combinations of optimal trajectories must be calculated to find the best one. This number of missions doubles to almost 19 million if there is no decision to return the samples together after collecting them or to drop each off at Earth after collection to mitigate risk. If a good educated guess can be made to limit the inclination and minimum size of the asteroid, this number of possible targets can be reduced to about 400[1]. This still leaves 160,000 missions to compare. Lastly, this methodology becomes unusable when the number of asteroids to be visited to increased from two. Just adding one more asteroid in the above scenarios increases the number of missions to 58 billion and 127 million respectively.

Another method is to treat this problem similar to a traveling salesman problem. This is a classical optimization problem to find the sequence of "cities" for a salesman to visit to minimize a cost variable, typically time. However, in this instance the fuel cost to travel between cities is unknown, so would need to be initially estimated. This method introduces discrete variables, i.e. the "cities" or asteroid orbits, and changes the nature of the problem to a mixed-integer nonlinear programming problem or MINLP. This method would depend on the accuracy of the ΔV approximation for the fuel cost for the transfer between each asteroid.

The final method would be to develop a method to solve a hybrid optimal control problem with integer decision variables, nonlinear dynamics, and discrete targets to find the optimal sequence, corresponding trajectories and maximum number of dynamic targets that could be visited for a given amount of fuel. This problem has not yet been solved and constitutes one of the grand challenges in mathematics.

---

[1] Based on assumptions given by JPL and then used to filter DASCOM asteroid database listed at http://ssd.jpl.nasa.gov/dastcom.html in March 2003.

THIS PAGE INTENTIONALLY LEFT BLANK

II. PROBLEM FORMULATION AND OPTIMAL SOLUTIONS

A. POLAR COORDINATE FRAME

A two-dimensional heliocentric polar coordinates frame, represented in Figure 1, will be used throughout this thesis and in all optimizations problems. This is convenient for low thrust trajectories since the transfer angle, or angular displacement state represented by the Greek symbol for nu, ν, is continually and steadily increasing from the starting value. Interplanetary low thrust missions typically consist of multiple revolutions and a quick look at the polar states can indicate which revolution and at what radius an event occurs. More importantly, the position states will vary slowly over the entire mission. In a Cartesian coordinate frame, the position states would be much more sinusoidal in nature and typically more difficult for discrete optimization methods to model without a higher corresponding number of discretization points and the associated computing time to prevent aliasing problems. Other coordinate elements, such as equinoctial elements, can provide a set of singularity-free slowly changing variables, but were not used largely due to the difficulty in finding errors in each problem mathematical definition and results since they have little direct physical interpretation.
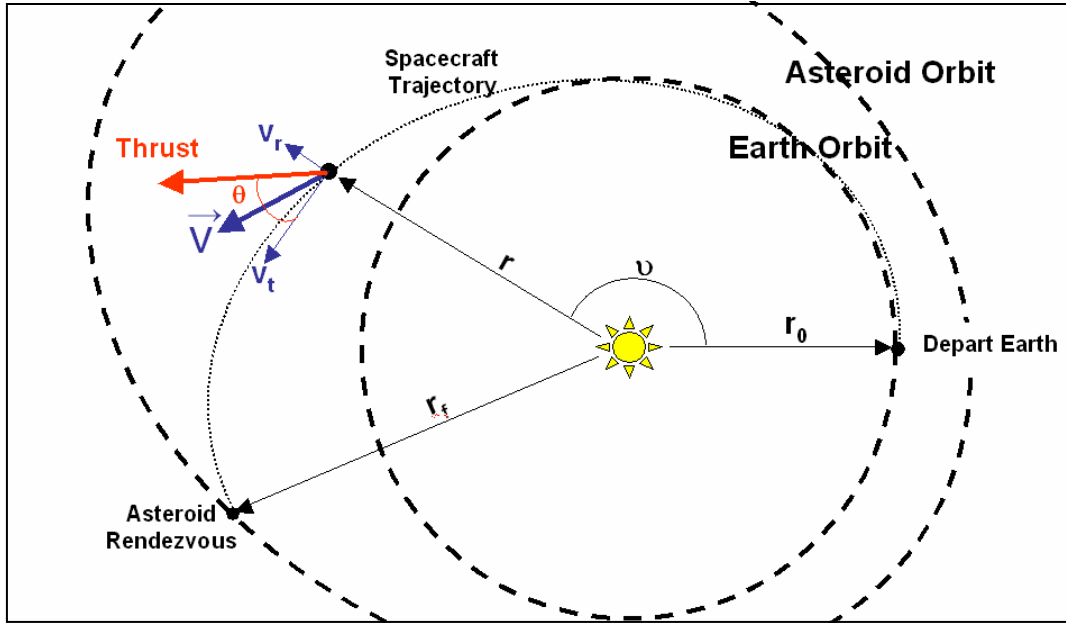
Figure 1   Polar Coordinate Frame

B. EQUATIONS OF MOTION

Using the above polar coordinates, the state vector $\mathbf{x} = [r, \upsilon, v_r, v_t, m]^T$ describes the spacecraft position, velocity, and mass states. The spacecraft control vector is $\mathbf{u} = [T, \theta]^T$, which describes the engine thrust magnitude and the angle from the local horizontal in which it is applied. The equations of motion are expressed as the function $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u})$ as follows:

$$\dot{\mathbf{x}} = \begin{bmatrix} \dot{r} \\ \dot{\upsilon} \\ \dot{v}_r \\ \dot{v}_t \\ \dot{m} \end{bmatrix} = \begin{bmatrix} v_r \\ \dfrac{v_t}{r} \\ \dfrac{v_t^2}{r} - \dfrac{\mu}{r^2} + \dfrac{T}{m}\sin\theta \\ \dfrac{-v_r v_t}{r} + \dfrac{T}{m}\cos\theta \\ -\dfrac{T}{Isp * g_0} \end{bmatrix} \tag{1-5}$$

where
r	$\equiv$ radial distance from sun
υ	$\equiv$ transfer angle or angular displacement
v_r	$\equiv$ radial velocity component
v_t	$\equiv$ transverse velocity component
m	$\equiv$ vehicle mass
μ	$\equiv$ gravitational parameter of sun
T	$\equiv$ thrust magnitude
θ	$\equiv$ thrust direction angle from local horizontal
Isp	$\equiv$ specific impulse of engine
g_0	$\equiv$ gravity constant at Earth surface

This dynamics model is a two-dimensional, two-body (sun and spacecraft) system with a $1/r^2$ gravity assumption. Third body gravity interactions or Earth atmospheric drag is not included in Equations 1 through 5. However, specific events such as propellant used at the asteroid, stay times at the asteroid, drop mass of sample at Earth flyby and Earth gravity assists are handled by the optimization routine as discrete events and it is not necessary to include them in the continuous equations of motion. The gravitational Sphere of Influence (SOI) for planets in the inner solar system and the moon are

relatively small compared to the scale of the system [Ref. 3]. For a reference, the SOI for Earth extends to just 6 thousandths of the average distance from the Earth to the Sun, or 1 AU. Thus, Earth and asteroids are just represented as point masses.

Even though three-dimensional trajectories are not optimized in this thesis, the utility of the developed methodology apply to higher fidelity dynamical models. Robert Stevens [Ref. 4] previously showed that the specific optimization code used, called DIDO, can simultaneously optimize the interplanetary extremely low-thrust rendezvous trajectory and return trajectories in all three dimensions, even with varying stay times at the target. Purposefully, the real effort was to develop methodologies in accurately determining reachable sets and not in finding the highest fidelity model possible to plan a specific low thrust trajectory.

C.   MISSION MODEL

A baseline of mission parameters where established by JPL. These parameters where:

- Spacecraft dry mass

- Power available at spacecraft end of life

- Engine selection

- Launch vehicle

- Earth departure hyperbolic excess velocity (C_3)

- Stay time at each asteroid

- Maximum Earth flyby velocity (V_∞ at arrival)

- Minimum and Maximum Earth flyby altitude

- Propellant used for sample collection activities at each asteroid

- Sample drop mass at Earth returns

- Engine duty cycle to facilitate communications

- Maximum total mission time

As previously stated, the purpose is not to figure out how much fuel is needed to return two samples to Earth from given targets, but to find which asteroids are feasible options, given a set of mission parameters. The fuel budget is driven by the spacecraft design and launch vehicle selection that is primarily determined by the available budget. Additionally, some of the above baseline parameters were chosen given acceptable ranges and the final values were determined by the optimization code. These opportunities to have a higher fidelity model will be explained in later chapters.

D. LAUNCH VEHICLE MODEL

For a given launch vehicle a tradeoff can be made between propellant mass and the characteristic energy, most commonly know as C_3. The C_3 defines the energy with which a spacecraft leaves a planet's SOI and is equal to the square of the velocity at "infinity", or V_∞, which is the velocity of the spacecraft in excess of the planet's heliocentric velocity. This tradeoff can be easily optimized in this formulation since the initial mass and velocities are states and can be adjusted. The initial radial and transverse velocities of the spacecraft can be computed using Equation (7).

$$V_e = \sqrt{\frac{\mu_s}{R_e}} \tag{6}$$

where
V_e ≡ Velocity of Earth (approximated to be only in tangential direction)
R_e ≡ Earth radius to sun

$$C_3 = V_\infty^2 = \left(\sqrt{v_{r_0}^2 + v_{t_0}^2} - V_e \right)^2 \tag{7}$$

This tradeoff can be done is in two ways. The DIDO optimization code can interpolate between neighboring values in the table or a continuous polynomial function can be created to represent the table. The spacecraft has a dry mass where there is no more fuel to trade off and thus the minimum C_3, 0 km^2/s^2, and maximum C_3, approximately 10 km^2/s^2 [Ref 5]. In practice, since the C_3 is related to spacecraft mass and this can change in each iteration until an optimal result is found, it saves computational time and stability to fit the launch vehicle data into a 5th order polynomial using the MATLAB "polyfit" function and then using the simple corresponding

"polyval" function in each iteration than doing a table interpolation with "interp1". Thus the relationship between the initial spacecraft mass and velocity states can be written as [Ref. 6]:

$$p_5 m_0^5 + p_4 m_0^4 + p_3 m_0^3 + p_2 m_0^2 + p_1 m_0 + p_0 = \left(v_{t_0} - 30.18 \frac{km}{s} \right)^2 + v_{r_0}^2 \qquad (8)$$

where
p_i $\equiv$ i[th] polynomial coefficient of launch vehicle performance
m_0 $\equiv$ initial propellant mass of spacecraft determined during optimization

For the work shown in this thesis, the launch vehicle model was not truly accurate but approximates within +/- 15% of a Delta II's estimated performance.

E.   SPACECRAFT PROPULSION MODEL

All the low thrust trajectories presented in this thesis use a Solar Electric Propulsion (SEP) propulsion engine model, specifically the NASA Solar electric propulsion Technology Applications Readiness (NSTAR) system. The electric power available to the engine from the power processing unit is a function of the solar array design, range to the sun, and solar array degradation model with time. This range of available electric power corresponds to a range of thruster performance characterized by efficiency, mass flow rate, specific impulse, and thrust. Also, due to the component designs in the SEP engine there is a hard minimum and maximum possible thrust of 19 and 92 mN's respectively. Parametric models, similar to those used in the Solar-Electric Propulsion Trajectory Optimization Program (SEPTOP) [Ref.s 5,6] were implemented to model the NSTAR performance (model Q). In order to compute the maximum power available to the thrusters, first the power available from the solar panel is computed in Equation (9) by multiplying the polynomial models for time (radiation) degradation and range to sun by the reference power for the Gallium Arsenide solar array panel. The distance to sun is not simply an inverse square law partly because solar panel output decreases as the temperature of the solar panel increases [Ref.s 5,6].

$$P = P_0 \bullet \left[\frac{ga_1 + \dfrac{ga_2}{r} + \dfrac{ga_3}{r^3}}{1 + ga_4 \bullet r + ga_5 \bullet r^2} \bullet \frac{1}{r^2} \right] \bullet \left(tcons_1 + tcons_2 e^{tcons_3 \bullet t} + tcons_4 \bullet t \right) \qquad (9)$$

9

where

P $\equiv$ power available from the solar array

P_0 $\equiv$ reference power at 1 AU and at beginning of life

ga_x $\equiv$ solar array distance model coefficients

        $\equiv$ {1.320770, -0.108480; -0.116650, 0.108430, -0.012790}

$tcons_x$ $\equiv$ solar array time degradation model coefficients

r $\equiv$ distance to sun

t $\equiv$ time

Once power available from the solar array is calculated, the spacecraft's housekeeping power is subtracted to leave the power available to the propulsion unit, as in Equation (10).

$$P_e = P - P_h \tag{10}$$

where

P_e $\equiv$ power of spacecraft available to propulsion units

P_h $\equiv$ housekeeping power requirements

This power available to the engines can be higher than the maximum power it was designed handle since most solar arrays are designed to provide the minimum acceptable power at end of life. Also, there is a minimum power required to run the power processing functions and set up the required electric field to acceleration the Xe ions. Thus, real constraints on the P_e minimum and maximum must be applied.

$$P_{e_{min}} \leq P_e \leq P_{e_{max}} \tag{11}$$

Due to the pointing angles required for communications, many spacecraft with ion engines do not thrust and communicate at the same time. Thus, a duty cycle assumption of the engines was determined to be 95% to allow and average of 5% of the time to be used for communications. Since the thrust arcs are measured in months and years, this can be just thought of as efficiency factor and exactly when the communications occur is not a large source of overall error. Now thrust (T) and mass flow rate ($\dot{m}$) can be calculated by the following fifth order polynomial models and the specific impulse (I_{sp}) can be found by [Ref. 6]:

$$T = \left(ct_1 + ct_2 \cdot P_e + ct_3 \cdot P_e^2 + ct_4 \cdot P_e^3 + ct_5 \cdot P_e^4 \right) \cdot duty\_cycle \tag{12}$$

$$\dot{m} = \left(cm_1 + cm_2 \cdot P_e + cm_3 \cdot P_e^2 + cm_4 \cdot P_e^3 + cm_5 \cdot P_e^4 \right) \cdot duty\_cycle \tag{13}$$

$$I_{sp} = \frac{T}{g_0 \bullet \dot{m}}$$ (14)

where

$\dot{m}$ $\equiv$ mass flow rate

ct_x $\equiv$ thrust coefficients for engine model (mN)

      $\equiv$ {0.26673e2; -.52301e2; .91639e2; -.3719e2; .52301e1}

cm_x $\equiv$ mass flow rate coefficients for engine model (g/s)

      $\equiv$ {0.25057e-5; -0.53539e-5; 0.62509e-5; -0.25364e-5; 0.36983e-6}

However, in reality the first simulations of spacecraft missions tend to precede detailed design of the spacecraft and crucial components, such as solar arrays. Since solar cell performance is constantly improving, the latest generation of cells is not wholly characterized for degradation over time in radiation environments and this means a simpler form of Equation (9) is required. Since the *tcons* could not be provided by JPL, the known end of life power for solar array can be substituted for multiplying the reference solar array power by a degradation model. Also, one difficulty with using polynomial-parametric models is that they can have more error at the maximum and minimum points. For instance, in Equations (12) and (13), it can be easily seen that for a zero P_e value, the result is a non-zero value for $T, \dot{m}$, and $I_{sp.}$ This requires some additional steps to ensure if $P_e = 0$, then $T, \dot{m}$, and $I_{sp} = 0$. Also, as can be seen in Figure 2 on the lower left mass flow rate graph, the slight dip around 1.9 AU is not a real engine artifact, but a polynomial modeling error.
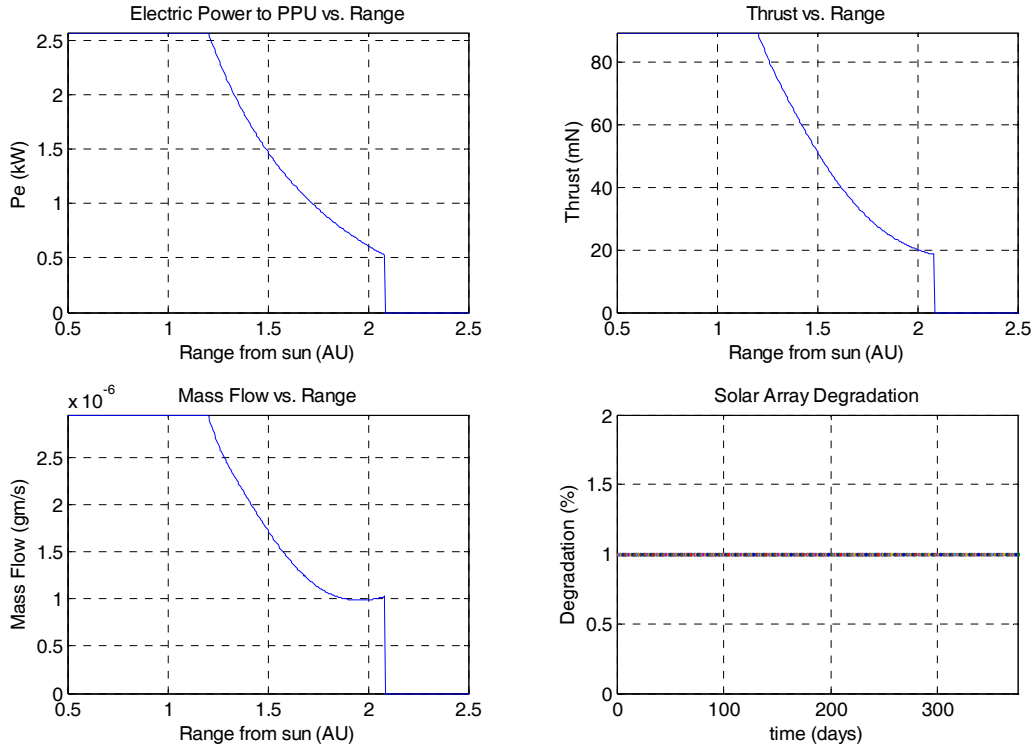
Figure 2   NSTAR Performance Model Using End-of-Life Power Vice Radiation
Degradation

Obviously, distance to the sun is the driving factor in the Ion Engine performance. However, since the *a* vs. *e* domain, or a 2-dimensional orbital element domain, will be extensively used throughout this thesis, the thrust performance is plotted in that domain in Figure 3. Since in an elliptical orbit ($e \neq 0$), the distance to the sun will vary from the perihelion range to the aphelion range of that orbit, the minimum and maximum available thrust varies and is referenced to as "best case" and "worst case". For example this graph clearly shows for this engine and solar array in an orbit with *a* = 1.5 and *e* = 0.3, the available thrust can be the engine's maximum at periapsis, but is zero at apoapsis. Thus, one could expect a coast phase in this orbit until the required engine performance is available to optimally maneuver the spacecraft. An artifact of the 5th order polynomial model mentioned previously near the minimum power points results in the artificial stair-step curves plotted in Figure 3.

Although not a subject for this thesis, this kind of plot can be used to balance the size of the solar array with the maximum range, fuel, or mission length of a vehicle. If P_e

is larger, then for the orbit in the previous example, the coast phase could be smaller and thus the total mission time would likely be reduced. An iterative systems engineering process can be done if a reachable target is near the lower thrust contours and thus any rendezvous could not be achieved near aphelion of the target.
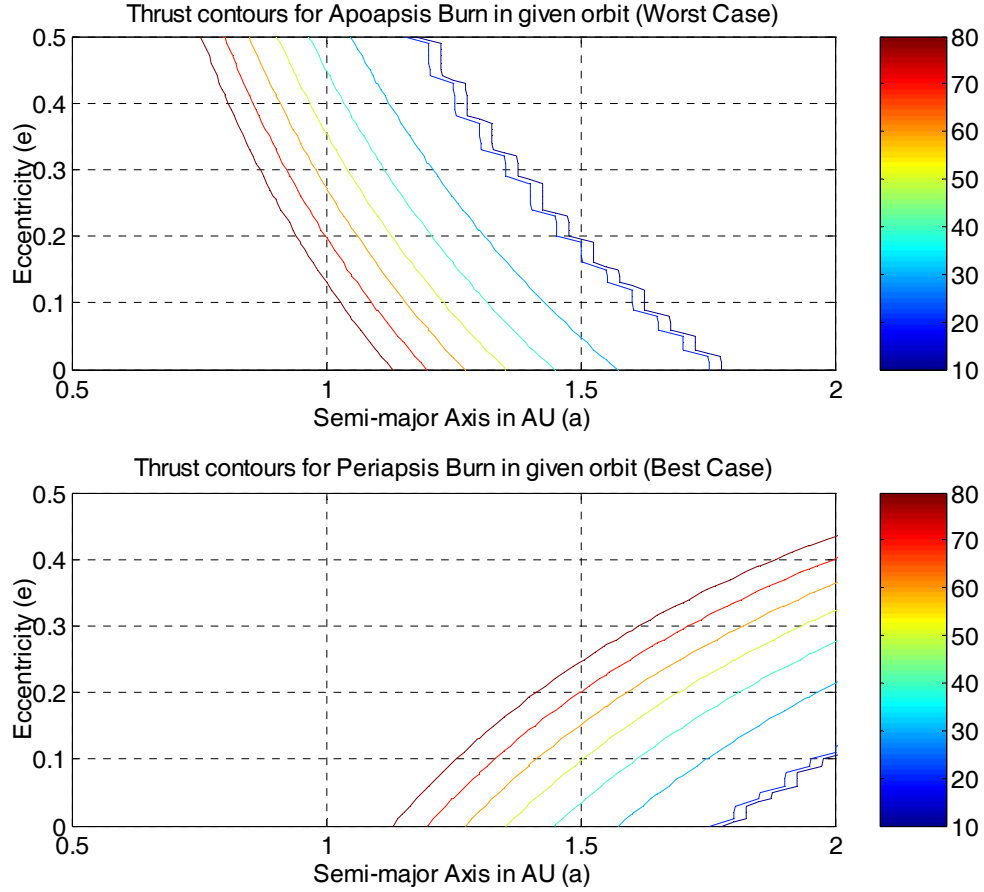
Figure 3   NSTAR Thrust Contours for Apoapsis and Periapsis Conditions (in mN)

F.   OPTIMAL CONTROL PROBLEM FORMULATION

The reachable sets will be constructed by solving multiple dynamic optimization problems where the objective function (*J*), also called the cost function or performance index, is extremized. A trajectory, or state-control function pair, results from solving for the maximum or minimum of an objective function is subject to a set of constraints to include the system's dynamical equations of motion. A generalized representation of this performance index, or cost function, is given by [Ref. 7]:

$$J\left[\mathbf{x},\mathbf{u},\tau_0,\tau_f\right]=E\left(\mathbf{x}(\tau_0),\mathbf{u}(\tau_f),\tau_0,\tau_f\right)+\int_{\tau_0}^{\tau_f}F\left(\mathbf{x}(\tau),\mathbf{u}(\tau),\tau\right)d\tau \tag{15}$$

where

$\mathbf{x}\equiv$ state variables (vector)

$\mathbf{u}\equiv$ control variables (vector)

$\tau_0\equiv$ initial time

$\tau_f\equiv$ final time

$E\equiv$ Event cost called Mayer Cost

$F\equiv$ Running cost or integral cost called Lagrange Cost

An "event" is the general term for any constraint, condition, or time that is at a boundary point or interior knots. In DIDO a knot is any point in a trajectory, or node, where constraints or conditions on the problem are imposed. Thus, the event times, τ_e, can be at any interior point, $\tau_0\le\tau_e\le\tau_f$. The even cost term, E, is commonly referred to as Φ in many papers and texts in optimal control theory [Ref. 2]. If used alone in the cost function, *J*, then the cost function can be termed as a Mayer Cost. This is the form that will be typically be used in throughout this thesis. The integral cost term, *F*, is commonly referred to as, *L*, in most papers and texts in optimal control theory and if used alone to define the performance index, then *J* can be called simply a Lagrange Cost. When both terms are used together, then the performance index is considered a Bolza Cost function.

The constraints for which the optimal solution of state-control function pairs must satisfy, includes the dynamic constraints such as Equations (1-5),

$$\dot{\mathbf{x}}(\tau)=\mathbf{f}(\mathbf{x}(\tau),\mathbf{u}(\tau),\tau) \tag{16}$$

the event constraints at end points and interior points,

$$\mathbf{e}_l\le\mathbf{e}(\mathbf{x}(\tau_0),\mathbf{x}(\tau_e),\mathbf{x}(\tau_f),\tau_0,\tau_e,\tau_f)\le\mathbf{e}_u \tag{17}$$

and the path constraints over the entire trajectories.

$$\mathbf{h}_l\le\mathbf{h}(\mathbf{x}(\tau),\mathbf{u}(\tau),\tau)\le\mathbf{h}_u \tag{18}$$

Path constraints are imposed across every node, such as thrust limits, vice event constraints that occur at specific points. Both LCDR Scott Josselyn and LCDR Robert

14

Stevens [Ref. 4,8] give clear explanations and examples of using these concepts in their theses.

If the path constraints only limit the value of a state or control between the two endpoints of a solution, then those path constraints can be more simply written as:

$$\mathbf{x}_l \leq \mathbf{x}(\tau) \leq \mathbf{x}_u \tag{19}$$

$$\mathbf{u}_l \leq \mathbf{u}(\tau) \leq \mathbf{u}_u \tag{20}$$

This type of path constraint is call a box constraint since it simple bounds the valid values for each state or control that is constrained in this manner. For example the a thrust direction can be constrained by $0 \leq \theta(\tau) \leq 2\pi$, however, using a SEP engine required a more complicated function where the thrust can not be as simply stated in one inequality constraint. Thus the path constraint for a high fidelity engine model must be a function, $h_i(\mathbf{x}(\tau), \tau)$.

Using Equations (16), (17), and (18) nearly all smooth dynamic optimization problems can be formulated in mathematical terms to be solve analytically or numerically.

G.   OPTIMAL CONTROLS SOLUTION METHODS

Numerical optimal control solvers can be generally classified into two categories, indirect and direct methods. Indirect methods use Euler-Lagrange differential equations and Pontryagin's Minimum Principle[2] (PMP) to formulate a complete set of state and costate dynamics and then discretize the system to solve for an optimal control history based on satisfying the first and second order optimality conditions. Direct methods transcribe an optimal control problem into a nonlinear programming problem (NLP) without using PMP or adjoint equations. NPLs are simply static optimization problems where the performance index or constraints are nonlinear functions [Ref. 9].

The problems presented here will be solved with a MATLAB application package created at the Naval Postgraduate School called DIDO that uses a pseudospectral method

---

[2] This theory was actually derived with the value of the Hamiltonian (*H*) to be of the opposite sign, which is common in classical literature. Thus the theorem is also called Pontryagin's Maximum Principle, which is the same as a minimum principle when *H* is defined with a positives sign as in modern western literature.

to solve smooth and non-smooth hybrid dynamic optimization problems. The state and control functions are discretized and collocated using Legrendre-Gauss-Labatto (LGL) points creating a carefully selected non-uniform grid between boundary points and/or any interior points. The solution method used by DIDO is based on pseudospectral approximation and "is significantly different from prior methods used to solve such problems and hence the code is a realization of a fundamentally new way of rapidly solving dynamic optimization problems." [Ref. 7] The resulting NLPs are large-scale and sparse so that only a small percent of the elements are nonzero. Thus SQP solvers such as SNOPT, which can be used by DIDO, can solve sparse problems over one hundred times faster than standard "dense" SQP methods [Ref. 10]. Thus, DIDO has proven to be a very robust and efficient optimization program that generally only requires crude guesses. Also, for problems without interior points, the costate dynamics can be approximated to demonstrate the optimality of the solution through DIDO's implementation of the Covector Mapping Theorem [Ref.s 5,11].

H. OPTIMALITY

The optimization process defines what is know as the "Hamiltonian" of the problem by combining, or "adjoining", the integral cost function in Equation (15) with the state dynamics in Equation (16) by use of Lagrange multipliers, $\boldsymbol{\lambda}$. [Ref. 2]

$$H\left(\mathbf{x},\mathbf{u},\boldsymbol{\lambda},\tau;\mathbf{p}\right) \equiv F\left(\mathbf{x}(\tau),\mathbf{u}(\tau),\tau;\mathbf{p}\right) + \boldsymbol{\lambda}^{\mathrm{T}}\mathbf{f}\left(\mathbf{x}(\tau),\mathbf{u}(\tau),\tau;\mathbf{p}\right) \tag{21}$$

where
$\mathbf{p} \equiv$ parameters (non-state, non-controls variables or constants)

These Lagrange multipliers are known as costates, or adjoints, and have and their own dynamics of,

$$-\dot{\boldsymbol{\lambda}} = \frac{\partial H}{\partial \mathbf{x}} \equiv \frac{\partial F}{\partial \mathbf{x}} + \left(\frac{\partial \mathbf{f}}{\partial \mathbf{x}}\right)^{\mathrm{T}}\boldsymbol{\lambda} \tag{22}$$

and from Equation (22) it can be easily seen that the number of costates is equal to the number of states. The transversality condition for the costate dynamics provides boundary conditions on the costate dynamics, given by;

$$\boldsymbol{\lambda}(\tau_o) = -\frac{\partial \overline{E}}{\partial \mathbf{x}_o} \text{ and } \boldsymbol{\lambda}(\tau_f) = -\frac{\partial \overline{E}}{\partial \mathbf{x}_f} \tag{23}$$

16

where

$\bar{E} \equiv E + \sigma\mathbf{e}$ = Augmented Events Funtion

$\sigma \equiv$ Lagrange multipliers for event contraints

or if the problem leaves time as a free variable then,

$$H(\tau_f) + \frac{\partial E(\tau_f)}{\partial \mathbf{x}_f} + \sigma^{\mathbf{T}} \frac{\partial e(\tau_f)}{\partial \mathbf{x}_f} = 0 \tag{24}$$

This new point cost uses another set of Lagrange multipliers, υ, to adjoin event constraints to the Event Cost, E. Now applying PMP to find necessary conditions for optimality, we also need,

$$\frac{\partial H}{\partial \mathbf{u}} = 0, \quad t_0 \le t \le t_f \tag{25}$$

and,

$$\frac{\partial^2 H}{\partial \mathbf{u}^2} \ge 0 \tag{26}$$

Equations (22), (23), and (25) are also known as the Euler-Lagrange equations in the calculus of variations [Ref.s 2,11]. Equation (26) is the second order condition for optimality and requires that the Hessian of the Hamiltonian is positive semi-definite.

However, if path constraints exist then an Augmented Hamiltonian must be formed, which adds the path constraints, $\mathbf{h}$, to the Hamiltonian by another set of Lagrange multipliers, μ, such as:

$$\bar{H}(\mathbf{x}, \mathbf{u}, \mu, \lambda, \tau; \mathbf{p}) \equiv H(\mathbf{x}, \mathbf{u}, \lambda, \tau; \mathbf{p}) + \mu^{\mathbf{T}}\mathbf{h}(\mathbf{x}, \mathbf{u}, \mu, \tau; \mathbf{p}) = F + \lambda^{\mathbf{T}}\mathbf{f} + \mu^{\mathbf{T}}\mathbf{h} \tag{27}$$

Now, the optimal control vector values, $\mathbf{u}*$ is found by minimizing H with respect to the control argument, $\mathbf{u}$, such that $\mathbf{u} \subset U$, where U is the domain of the control vector with respect to all the path constraints, or

$$\min \bar{H} \text{ wrt } \mathbf{u}, \text{ such that } \mathbf{h}_{lower} \le \mathbf{h}(\mathbf{x}, \mathbf{u}, \tau) \le \mathbf{h}_{upper} \tag{28}$$

This is the full, generalized form required with inequality constraints on the path.

1. Local and Global Optimality

The Hamiltonian may be a very complex function with local minimums and maximums. For each local minimum that satisfies all constraints, it will have a scalar cost function, *J*, value associated with that minimum. Of course, even in a limited domain of U, where control values satisfy the path constraints, there can be a huge number of local minimums. This leads into several problems for NLP optimization using numerical techniques. First, during search strategies for these local minimums, the global minimum may not be found due to approximation errors in discretizing the problem or many other errors in optimization methods. Second, when comparing local minimums, the difference in cost between nearby solutions may be below the level of error in determining the cost. Since, each local minimum solution, with its corresponding trajectory and control history, can satisfy the Euler-Lagrange equations and be locally optimum over a range of solutions, only can the simplest problem be solved for a globally optimal solution. Thus, all trajectories herein will be considered locally optimal and not discount the existence of a better solution. This is one reason why improving numerical techniques that provide the best approximation of the continuous system dynamics and other constraints with the least sensitivity to the initial guess of a solution is important.

2. Checking Optimality of Solutions

There are several properties of an optimal solution, i.e. $\mathbf{u}*$ which minimizes $\bar{H}$, that can be verified to check whether the DIDO output is indeed optimal. $\bar{H}$, known as the augmented Hamiltonian, or sometimes as the Lagrangian, must satisfy the Karush-Kuhn-Tucker (KKT) theorem conditions below.

$$\frac{\partial \bar{H}}{\partial \mathbf{u}} = 0, \quad t_0 \le t \le t_f \tag{29}$$

$$\boldsymbol{\mu}(\tau)^{\mathrm{T}} \mathbf{h}(\mathbf{x}(\tau), \mathbf{u}(\tau), \tau) = 0 \tag{30}$$

$$\mu_i(\tau) = \begin{cases} \le 0 & h_i(\tau) = h_l \\ \ge 0 & \text{if} \quad h_i(\tau) = h_u \\ = 0 & h_l < h_i(\tau) < h_u \\ any & h_l = h_u \end{cases} \text{,where} \tag{31}$$

18

where

$\mu_i = $ i[th] Lagrange multiplier for i[th] path constraint

$h_i = $ i[th] path constraint

$h_l = $ lower constraint for i[th] path constraint

$h_u = $ upper constraint for i[th] path constraint

These KKT conditions are shown to be satisfied by constructing a switching function for each path constraint to demonstrate that Equation (31) is satisfied. In the third case of Equation (31), where the constraint can be throttled between the extreme limits then at these times the augmented Hamiltonian is the same as Hamiltonian and thus Equations (29) and (25) are identical and Equation (26) can be verified to be true.

Another verifiable condition of optimality is the check that the Hamiltonian is a constant for all time since [Ref. 11]:

$$\text{given } H \equiv H\left(\mathbf{x}, \mathbf{u}, \lambda, \tau; \mathbf{p}\right)$$

now differentiating H with respect to time provides:

$$\frac{dH}{d\tau} = \left(\frac{\partial H}{\partial \lambda}\right)^{\mathrm{T}} \dot{\lambda} + \left(\frac{\partial H}{\partial \mathbf{x}}\right)^{\mathrm{T}} \dot{\mathbf{x}} + \left(\frac{\partial H}{\partial \mathbf{u}}\right)^{\mathrm{T}} \dot{\mathbf{u}} + \frac{\partial H}{\partial \tau} \tag{32}$$

and for $\mathbf{u}^*$, $\dfrac{\partial H}{\partial \mathbf{u}} = 0$ by PMP,

and $H = F + \lambda^{\mathrm{T}} \mathbf{f} \implies \dfrac{\partial \mathrm{H}}{\partial \lambda} = \mathbf{f} = \dot{\mathbf{x}}$

and costate dynamics are $-\dot{\lambda} = \dfrac{\partial \mathrm{H}}{\partial \mathbf{x}}$, then by substitution

$$\frac{dH}{d\tau} = \left(\frac{\partial H}{\partial \lambda}\right)^{\mathrm{T}} \dot{\lambda} + \left(-\dot{\lambda}\right)^{\mathrm{T}} \frac{\partial H}{\partial \lambda} + \left(0\right)^{\mathrm{T}} \dot{\mathbf{u}} + \frac{\partial H}{\partial \tau} \tag{33}$$

$$\text{thus, } \frac{dH}{d\tau} = \frac{\partial H}{\partial \tau} \tag{34}$$

Thus, if the Hamiltonian is not an explicit function of time, i.e. time is not a performance index or part of the dynamic constraints, then the Hamiltonian should be

equal to zero for all time. If the problem is a minimum time problem, the Hamiltonian would be equal to –1 or some other constant if the problem is time constrained.

To check these necessary conditions of optimality, the time history of the costates must be known. Since direct optimization methods do not explicitly program the costates to be solved simultaneously with the state dynamics, the CMT can be used to derive these approximate values at the collocation points for analysis. Currently, DIDO will provide the costates, $\lambda(\tau)$, for problems without interior event constraints [Ref. 7]. Therefore, only the solutions from Part A of the methodology here will demonstrate the optimality of trajectories computed. Future version of DIDO will provide the costates for problems with interior point constraints [Ref. 12].

3. Comparing Results with Known Optimal Solutions

The first trajectory examined in this thesis will be a low thrust transfer from a circular Earth orbit (radius = 1 AU) out to the maximum circular orbit distance from the sun for a given amount of fuel. This is a simple coplanar, circular-to-circular orbit transfer and optimal trajectories can be readily calculated. A first order analysis can be done using Edelbaum's [Ref. 13] analytical ΔV equation for constant acceleration circle-to-circle transfers. The algorithm assumes that thrust magnitude and spacecraft mass are both constant during the transfer and that the orbit remains or is forced to be circular after each revolution. Given the initial and final circular orbit velocities and $0 < \Delta i < 114.59°$, the initial thrust vector yaw angle is computed as:

$$\tan \beta_0 = \frac{\sin\left(\dfrac{\pi \Delta i}{2}\right)}{\dfrac{V_0}{V_f} - \cos\left(\dfrac{\pi \Delta i}{2}\right)} \tag{35}$$

This β_0 term is the angle between the thrust vector and the local horizontal direction. Then the total velocity change for the low-thrust orbit transfer is:

$$\Delta V = V_0 \cos \beta_0 - \frac{V_0 \sin \beta_0}{\tan\left(\dfrac{\pi}{2} \Delta i + \beta_0\right)} \tag{36}$$

For the coplanar transfer case, this simplifies to:

$$\Delta V = \left| V_0 - V_f \right| \tag{37}$$

Another simple verification is to compare the solution with the optimal Hohmann Transfer for impulse coplanar, circle-to-circle orbit transfers. If time is not a constraint in the low-thrust optimal control problem, then one possible answer is a near infinite amount of very short pulses at perigee and in half a revolution another short pulse to re-circularize the orbit in each revolution. This would be a Hohmann approximation for a low thrust transfer. The Hohmann Transfer can be computed as follows [Ref. 14]:

$$a_{trans} = \frac{r_0 - r_f}{2}$$

$$V_{trans_a} = \sqrt{\frac{2\mu}{r_0} - \frac{\mu}{a_{trans}}} \qquad V_{trans_b} = \sqrt{\frac{2\mu}{r_f} - \frac{\mu}{a_{trans}}} \tag{38}$$

$$\Delta V = \left| V_{trans_a} - V_0 \right| + \left| V_f - V_{trans_b} \right|$$

The first scenario later examined will be a simple coplanar orbit transfer trajectories to rendezvous with an asteroid and thus the previous two known optimal solutions can be reasonably compared with the result to again check the optimality of the DIDO computations.

I.   FEASIBILITY

Another check of a solution of an optimal control problem is that the trajectories and control histories are feasible. This means that the control histories can be applied to the starting point (states at initial boundary) and reach the end point (final boundary) using the system dynamics and meeting every constraint and event specified to a high degree of accuracy. In other words, is the solution real and did it meet all the criteria specified.

1.   Propagating the Optimal Control History, u*

Given an optimal control vector, $\mathbf{u}(t)^*$, these controls can be applied to the system dynamical constraints given in Equation (16), or the equations of motion and for the initial state vector, $\mathbf{x}^0$, the trajectory should follow the state history predicted in the solution, $\mathbf{x}(\tau)$, and end up at the final state vector, $\mathbf{x}^f$. If this is not true, then the solution is infeasible and not valid for the given control history. Additionally, if any

constraints, such as upper or lower limits on states or controls or events, are not maintained then the solution is also not valid. This is relatively straightforward since the solution and the dynamics all known and one must just check that the optimal controls can propagate to the desired, or given, final state. The difficulty is to figure out how to propagate the Ordinary Differential Equations (ODEs) represented by Equation (16) and how to judge if the result is close enough to the state history predicted by the optimization code.

The propagation methods used herein are described in more detail in Appendix A [Ref. 15]. They represent a broad spectrum of numerical theory to solve ODEs and are implemented in MATLAB. However, any mathematical program should be able to implement and duplicate the results from any type of ODE solver chosen. Since MATLAB has seven ODE solvers from which to choose, all are used to propagate the control histories and a total error is computed for each ODE solver. Then the solver with the least error is selected to plot and analyze with the output of DIDO. So far no single solver seems to be the best candidate to choose to propagate for all solutions.

2. Error in a Propagated Solution

Since any numerical solution is given in discrete values at discrete time intervals and only approximate a continuous solution, errors will be present. Since exact analytical solutions are not existent for these problems, defining from what standard the error will be measured and how can be subject to some debate [Ref. 12]. Recall the purpose is to evaluate if the control history that the DIDO code determines to be optimal is in fact feasible and to also be determine which of the seven available ODE solvers is the best performing for a particular optimal control and state response. Thus, we will consider the exact solution to be the DIDO result and evaluate each of the ODE propagated results to be approximates. Thus, the Euclidean L2 integral root mean square (RMS) error norm is computed by:

$$\mathbf{e}_i = \mathbf{\Psi}_i - \hat{\mathbf{\Psi}}_i \tag{39}$$

$$\mathbf{e}_{L2} = \sqrt{\frac{1}{\mathbf{\Psi}^2_{max}} \sum_{i=1}^{n} \mathbf{e}_i^2} \tag{40}$$

where

$\mathbf{e}_i \equiv$ vector of state errors at node i

$\mathbf{e}_{L2} \equiv$ Euclidean L2 integral RMS error norm

$n \equiv$ number of LGL nodes or max(i)

$\mathbf{\Psi}_i \equiv$ optimization code solution state vector at node i

$\hat{\mathbf{\Psi}}_i \equiv$ propagated solution state vector at node i

$\mathbf{\Psi}_{max} \equiv$ maximum value of optimized solution for each state

Since the state and control history is evaluated at the LGL points the difference between the DIDO result and a propagated result RMS error is compared at these points. Figure 4 is a sample MATLAB script output generated from the error computation code:

```
==================================================
Propagator Performance in order best to worst
==================================================
State  Prop'r       Prop'r End  DIDO End  Total Error Norm
--------------------------------------------------------------------
  r     ode23tb      1.65        1.651     0.0207
  r     ode23s       1.653       1.651     0.023
  r     ode113       1.650       1.651     0.033
  r     ode23        1.653       1.651     0.0491
  r     ode23t       1.642       1.651     0.0816
  r     ode45        1.643       1.651     0.142
  r     ode15s       1.653       1.651     0.145
ode23tb selected
```

Figure 4   Error Comparisons

The radial distance state is shown for example and from this comparison, the ODE solver 23tb, a Runge-Kutta method where the first stage uses a trapezoidal rule step and the second stage is a backward differentiation, has the least accumulated error [Ref. 15]. This ODE propagated result is then used for all other plots and analysis. A different comparison between the available ODE solvers to examine the deviation between all 7 ODE solutions to see which states vary the most between the different solvers. This result is plotting in Figure 5 and the results are nearly identical for every simulation; the transfer angle, or true anomaly (in radians), has the most error than any other state. Of course, the deviations are normalized by the maximum value in each of the state histories so that units are not a factor in this comparison.
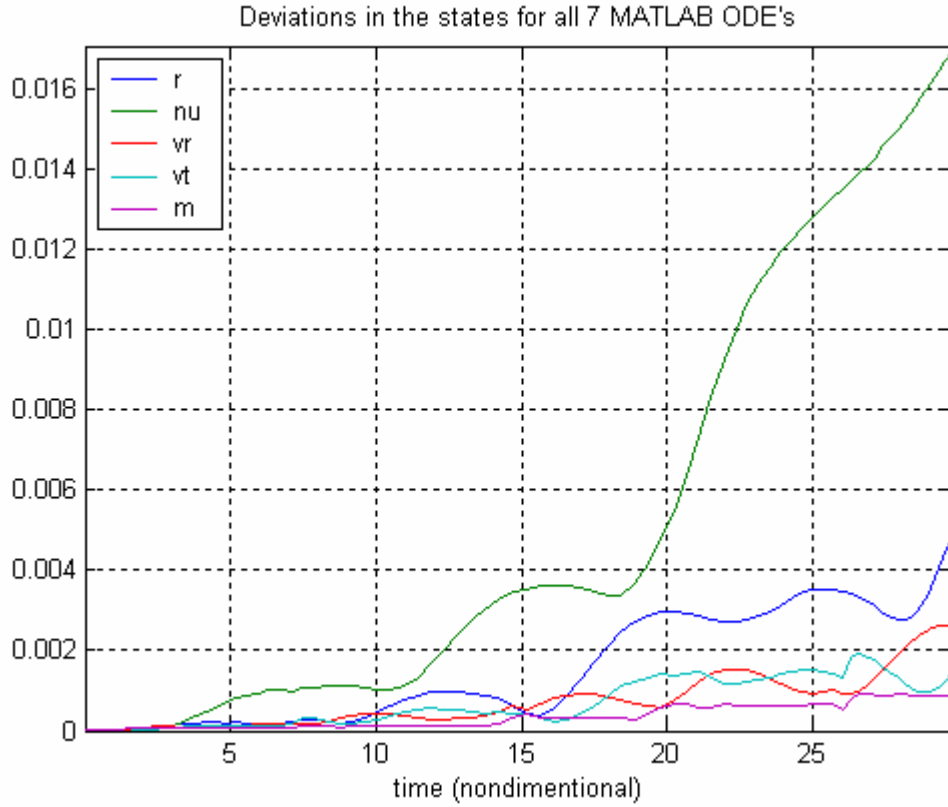
Figure 5   State deviations between ODE solvers over time

Finally, a typical trajectory that had a good feasibility check between the DIDO optimal solution and the best propagated solution can be seen in Figure 6. In this figure, the circles represent 200 nodes where the DIDO optimization code computed for the optimal state history and the line is drawn from the propagated solution using the control history from DIDO. When the line passes through the middle of the circle, this is the best result possible. Small arrows are also plotted to represent the thrusting and direction. The total error is from the previous example and equal to 0.0207.
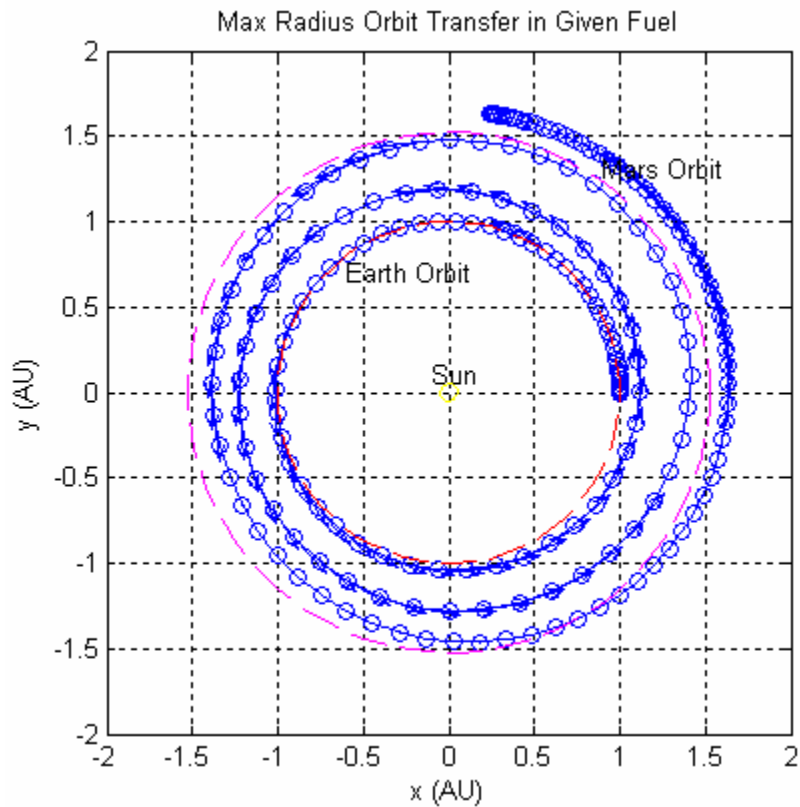
Figure 6   Good Feasibility Check between DIDO and Propagated solution

A poor correlation between the DIDO result and propagated solution ensures that an error in the optimization code does not result in an optimal control history that is unexecutable. Figure 7 shows an example where DIDO may return an answer, but it does not match the propagated solution and would typically violate boundary conditions or constraints. Again, the circles are the DIDO solution and the line is the propagated solution. In this case the total error is on the order of 10 or more and can easily be identified as not feasible and thus not optimal.
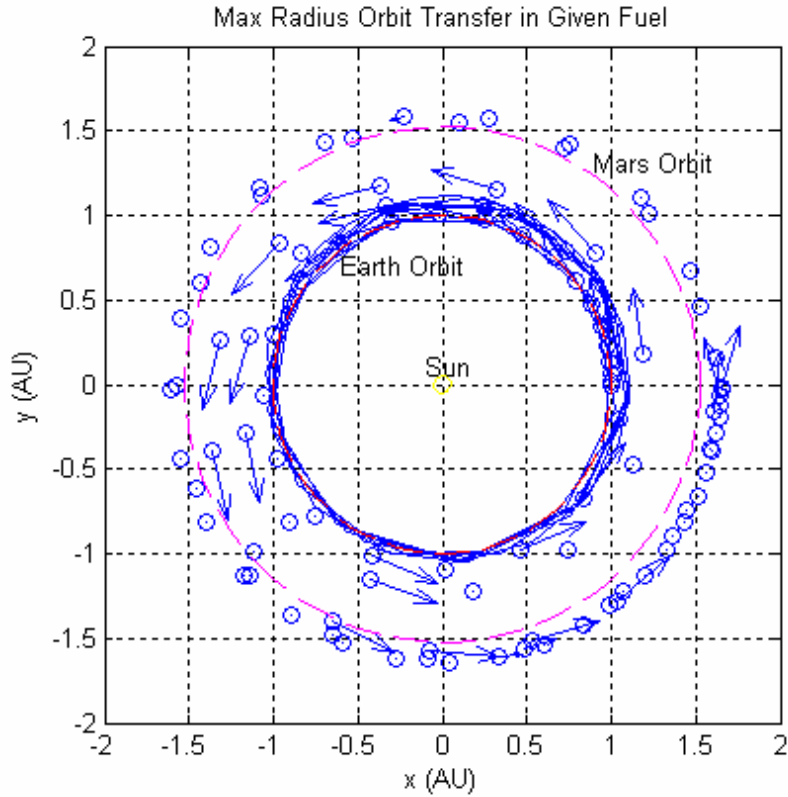
Figure 7   Not Feasible Solution

The most difficult results to interpret were the solutions where the error turned out to be on the > 0.2 and < 1.5. An example of this possibly feasible result is shown in Figure 8. In cases such as these, there is typically better ways to formulate the problem. In this particular case, a "no guess" startup was used with 80 nodes. This means that only the state start and endpoint guesses were made without even the benefit of reasonable assumptions (entered only since required by DIDO software). For example the ending radius was guessed to be 5, when it turned out to be about 1.65, or over 200% off the mark.
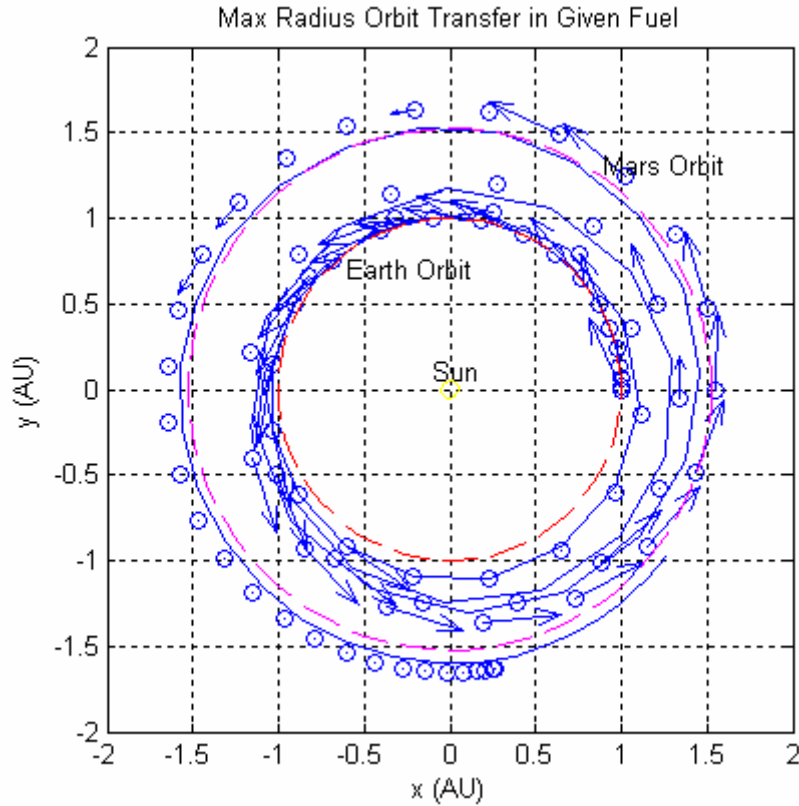
Figure 8   Possibly Feasible Result

Now, when the same problem is rerun with the this poor result starting and ending states "bootstrapped" into the guess of the next run, the result is much more favorable to determining feasibility. Figure 9 shows the benefits of at least making a fair guess at the endpoints, even it there is no guess for the 78 nodes in the middle. The runtime was reduced by over 28% by this simple act of having a good guess at the endpoint.

Now if the entire state history of the "no guess" solution from Figure 8 is bootstrapped to an 80 point guess for rerunning the same problem, the result is shown in Figure 10. Not only is the solution again easily recognizable as feasible, but the runtime is reduced by over 50% (from 4.2 minutes to 2.0 minutes). This type of benefit to bootstrapping has led DIDO users, notably CAPT Jon Strizzi, to reduce the overall run time by first running a low node (about 20) "no guess" problem and then without any analyzing immediately bootstrapping the state and control history to the guess of an 80 node trajectory optimization run. Thus, DIDO can easily and automatically reduce the run time and still literally use a random number guess to initialize.
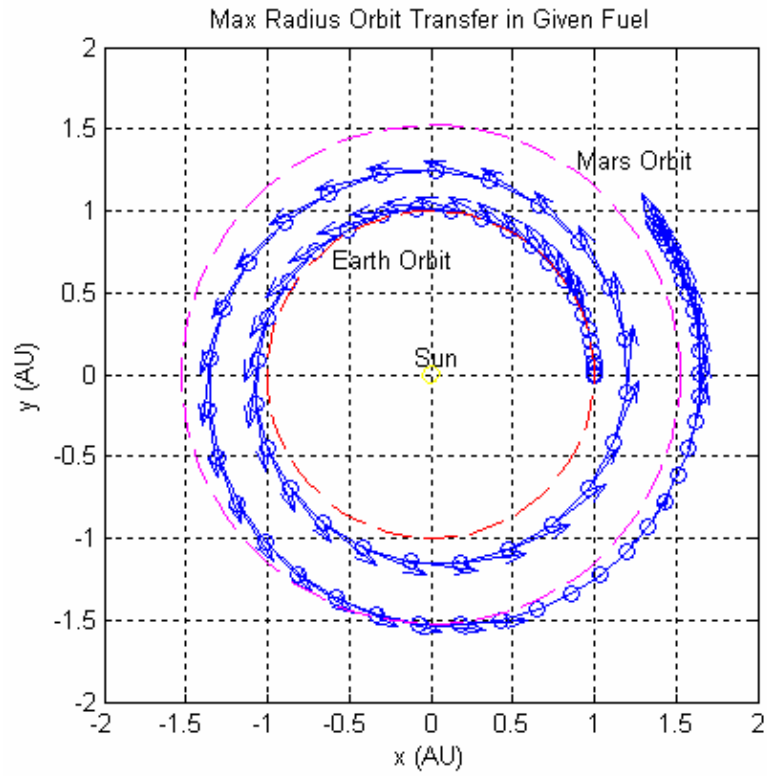
Figure 9   Feasible Result by Bootstrapping
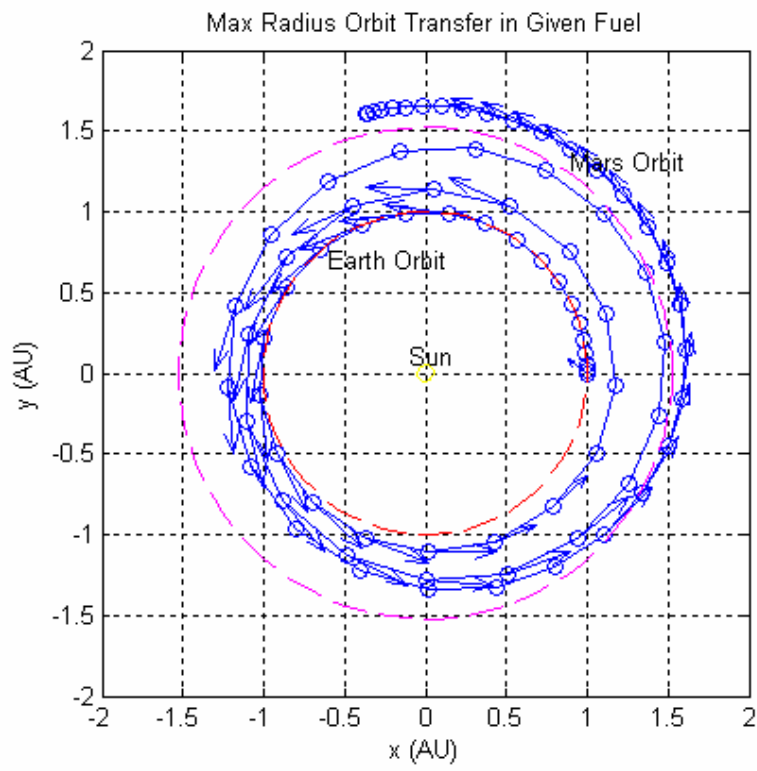
Figure 10   Feasible Result by using full Bootstrapping

Note that this is a case where there are many locally optimal solutions for this maximum radius optimal trajectory run. Figure 6, Figure 8, Figure 9, and Figure 10 are the same problem with different trajectories that are optimal and reach the same final radius (1.65 AU); however the control histories are not the same.

3. Check Results with Ideal Rocket Equation

In order for the resulting trajectory to be feasible and optimal, then the fuel must be completely used and the total ΔV imparted on spacecraft can not exceed the theoretical maximum. The Ideal Rocket Equation [Ref. 16] computes the maximum vehicle velocity change.

$$\Delta V = g_0 I_{SP} \ln\left(\frac{m_0}{m_0 - m_{prop}}\right) \tag{41}$$

This can be compared to the total velocity change in a given trajectory by integrating the acceleration over the entire trajectory to ensure the two previously mentioned requirements are met.

$$\Delta V = \int_{t_0}^{t_f} Accel_i dt = \int_{t_0}^{t_f} \frac{T_i}{m_i} dt \tag{42}$$

Thus it is necessary, but not sufficient, that for a feasible and optimal solution the computed result of Equation (42) should be no more than and very close to the result of Equation (41).

J. NON-DIMENSIONAL SCALING

In general, solving optimal control problems becomes faster and more stable when setting up the problem with well-ordered numerics. This statement is more or less true depending on the nature of the problem and solution methodology. Additionally, the use of non-dimensional units many times helps with quickly understanding a problem. For instance, when compute a radial distance from the sun of 156,333,934 km it takes some effort to compare it to the Earth's average distance from the sun of 149,597,870 km. However, if a distance unit is set where the radius of the Earth to the sun (R_e) is 1 distance unit, then our computational result would be an easily recognizable 1.045 distance units. In this simple example the distance unit coincides with the definition of Astronomical Units (AU). With computers, simple conversions can be done and displays

can be manipulated to show results in any format desire, thus the really need of non-dimensionalization is due to the numerical stability in the result.

For any set of new basic units, only three primary units must be defined: a distance unit, a time unit, and a mass unit. For this thesis, a form of heliocentric canonical units will be used. Fist the distance unit, U_{dist}, defined as one Earth radius. Then the time unit, U_{time}, is defined as the value that will set the sun's gravitational parameter, μ_{sun}, to be unity (equal to about 58.1 days):

$$U_{time} = \sqrt{\frac{U_{dist}^3}{\mu_{sun}}} \tag{43}$$

The mass unit, U_{mass}, fairly arbitrary for heliocentric canonical units, so it will be set so that the spacecrafts maximum wet mass will be unity. This will then be used to derive the velocity units, acceleration units, force units, energy units, and power units that have mass terms in their computations as follows [Ref.s 8,11]:

$$U_{vel} = \frac{U_{dist}}{U_{time}}$$

$$U_{accel} = \frac{U_{dist}^2}{U_{time}}$$

$$U_{force} = \frac{U_{mass} \bullet U_{dist}^2}{U_{time}} \tag{44-48}$$

$$U_{energy} = \frac{U_{mass} \bullet U_{dist}^3}{U_{time}}$$

$$U_{power} = \frac{U_{mass} \bullet U_{dist}^3}{U_{time}^2}$$

The scaling can be done in any order as long as a mass unit, time unit, and distance unit is uniquely defined, in effect creating new unit system. For instance, if a particular force and velocity term needed to be scaled to closer to unity value, then those units can be set. Then after a mass unit is determined, all the others can be solved. The following table lists the non-dimensional heliocentric canonical scaling used in this thesis and some of the normalized constants.

30

	Unscaled Units	Conversion Factor	Value
Time	s	U_{time}	5.0226×10^6
Distance	km	U_{dist}	149.597870×10^6
Mass	kg	U_{mass}	1273
Velocity	km/s	U_{vel}	29.7847
Acceleration	km²/s	U_{accel}	5.9301×10^{-6}
Force	kg-km²/s	U_{force}	0.0075

Table 1.   Non-dimensionalization Values

Now these new unit definitions can be used to transform and constants or variables in the problem to a non-dimensional and better-behaved set of dynamical numbers. For instance, in Equation (5) the exhaust velocity, or $v_e = I_{sp} \times g_0$, would normally be computed by multiplying the specific impulse by the gravitation constant and has units of meters per second ($s \times m/s^2 = m/s$). For example, to make this constant non-dimensional it can be converted term by term or in whole:

$$v_e = I_{sp} \bullet g_0 = 3113s \bullet 9.81\frac{m}{s^2} = 30538.53\frac{m}{s}$$

$$\overline{v}_e = \frac{v_e}{U_{vel}} = \frac{I_{sp}}{U_{time}} \bullet \frac{g_0}{U_{accel}} = 1025.31$$

(49)

The bar over any symbol will be used to denote that the appropriate non-dimensional units scaled the original symbol's units. Looking back at Equation (5), in order to make it non-dimensional then Equation (30) could be done by substitution:

$$\overline{m} = \frac{m}{U_{mass}}; \quad \overline{t} = \frac{t}{U_{time}}, \text{ thus}$$

$$\frac{d\overline{m}}{d\overline{t}} = \frac{U_{time}}{U_{mass}}\frac{dm}{dt} = \frac{U_{time}}{U_{mass}}\left(-\frac{T}{Isp * g_0}\right)$$

(50)

In a real example problem using typical values in this thesis, the maximum mass flow rate is -3.0126e-006 kg/s (approximately a loss of 3 milligrams a second). Now, after the non-dimentionalization in Equation (31), the maximum mass flow rate would be -0.0151. This new number is approximately four orders of magnitude closer to unity than the SI units result and thus more balanced. One could also observe that to solve the balancing problem in Equation (5), we can redefine the problem using milligrams vice kilograms and SI units. However, this would just cause stability problems elsewhere, such as making the starting mass state become 1,000,000 vice 1.

The other dynamics in Equations (1-4) can be similarly converted for non-dimensional computations using the similar form of:

$$\dot{x} = f(x,u,t) \tag{51}$$

$$\bar{\dot{x}} = \frac{U_{time}}{U_{x-units}} f(x,u,t) \tag{52}$$

Since the state dynamics are computed within the optimization code using non-dimensional units, all the states and control guesses, boundaries, constraints, and costs must be similarly non-dimensionalized. This method is using unscaled dynamics, since when computing the state derivatives, the states, controls, time or any constants must be unscaled into their original units, then Equation (51) computed, and finally Equation (52) used to convert back to the scaled units. Thus, this is considered using *Unscaled Dynamics*.

Another method is to scale all the states, controls, and constants before doing any optimization and only unscale after all the optimization calculations are performed. Thus, the method used in Equation (53) is called, *Scaled Dynamics*. In this method, the entire setup is using numbers already scaled and demonstrated in the example problem of the DIDO User's Manual [Ref. 7].

$$\bar{\dot{x}} = f(\bar{x},\bar{u},\bar{t};\bar{p}) \tag{53}$$

In this way, everything from start to end is done in the non-dimensional space in units that are predefined and arbitrary as long as they result in stable solutions. A comparison is shown in the Figure 11 and Figure 12.
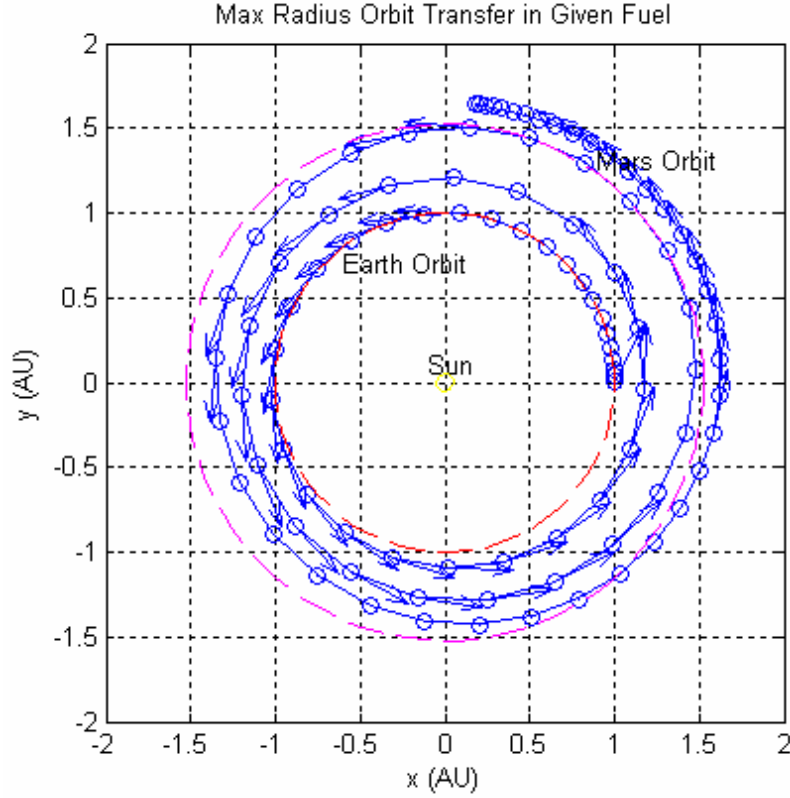
Figure 11   Results with Unscaled Dynamics from Equation (52)

Both methods produce feasible and optimal results that are very similar. Using the Scaled Dynamics method took almost 20% longer to run, even with fewer computations in the dynamics file (unscaling and rescaling). However, the final DIDO computed radial distance was within 0.06% of the propagated result, whereas the Unscaled Dynamics result was within 0.55% of the propagated radius state. This may have resulted from the n-state unscaling and rescaling operations add some computational machine error at each node for each iteration of the optimization routine. Therefore, even though this comparison was not very robust, the Scaled Dynamics of Equation (53) will be used throughout the rest of this thesis.
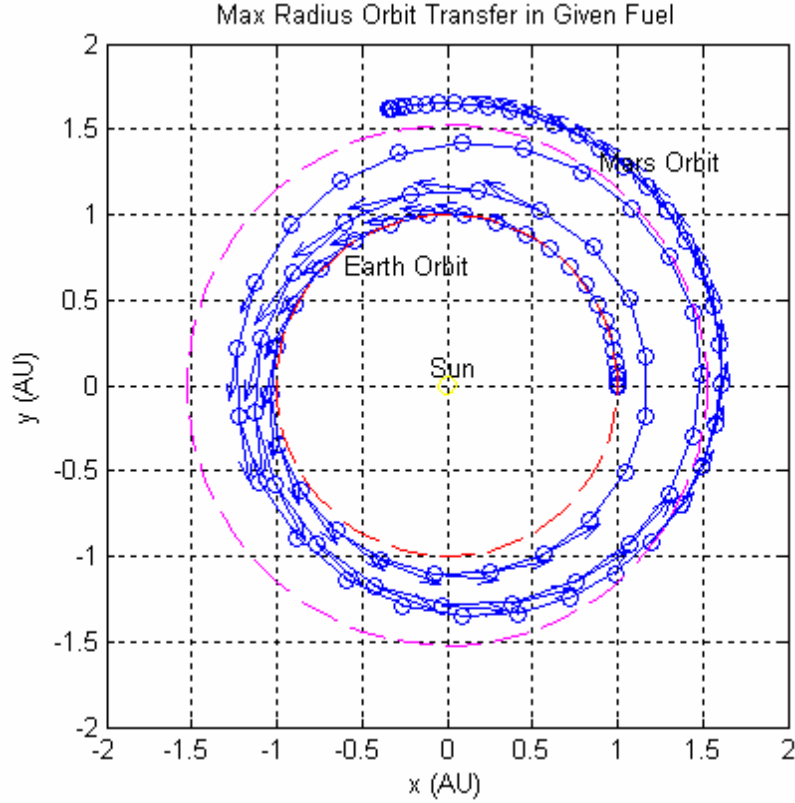
Figure 12   Results with Scaled Dynamics from Equation (53)

K.   BALANCING

Exceptionally large or small numbers can result in instability for the methods used by DIDO.   Thus, once the problem is favorably non-dimensionalized, each state, boundary condition, dynamical equation results (state dots), and starting and endpoint conditions should be checked to ensure that the range is closest possible to unity values. This process is called "balancing" and was tried in thesis for the Thrust state with no impact and so not utilized.

After the scaling in the previous section, the results will be evaluated to look for terms that may need some balancing.   For better graphical representation, the most significant parts of the script output of Appendix B is graphed in Figure 13, Figure 14, and Figure 15. These figures show the bounds (limits) imposed on the states and controls and the actual minimum and maximums for the states, controls, and derivatives of the states.   A well formulated problem (for DIDO software) will give the program some latitude in the possible values of the states and controls, even if these values are beyond a

reasonable value. However, well balanced problems have all the variables to a small, but discernable, order of magnitude and are values near each other. For instance, in Figure 13, the transfer angle varies from 0 to almost 14 units compared to the radial distance, which varies from 1 to 1.6. This one order of magnitude difference is the states are relatively small compared to the unscaled km units which would be nine orders of magnitude larger than the scaled distance units.

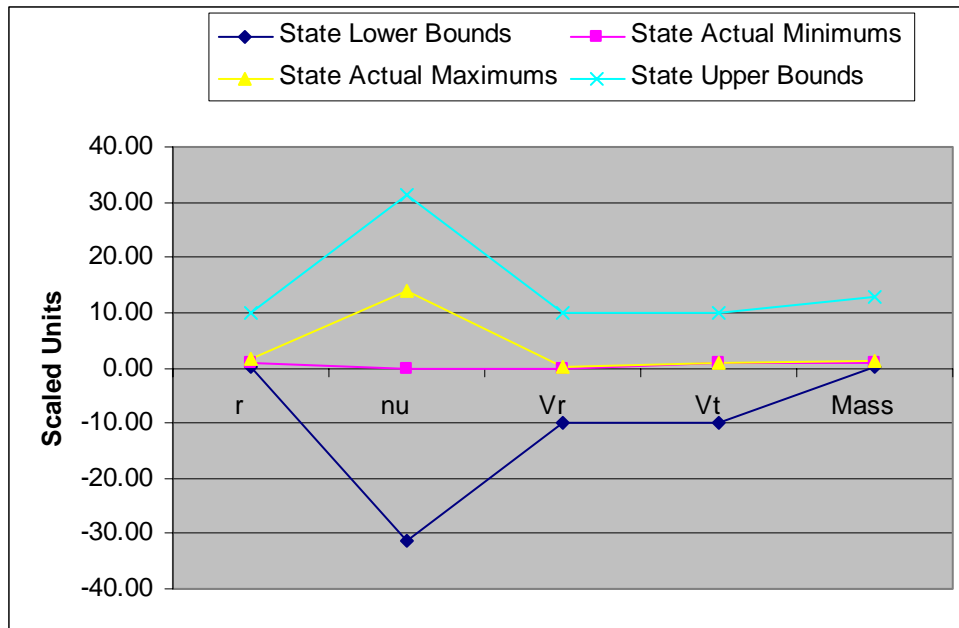Figure 13   State Limits and Actual Ranges
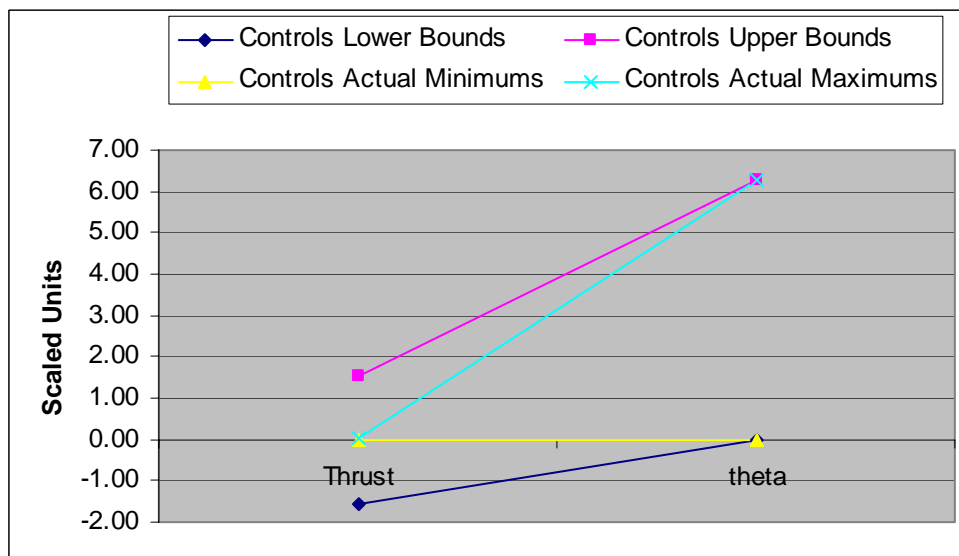
Figure 14   Control Limits and Actual Ranges

35
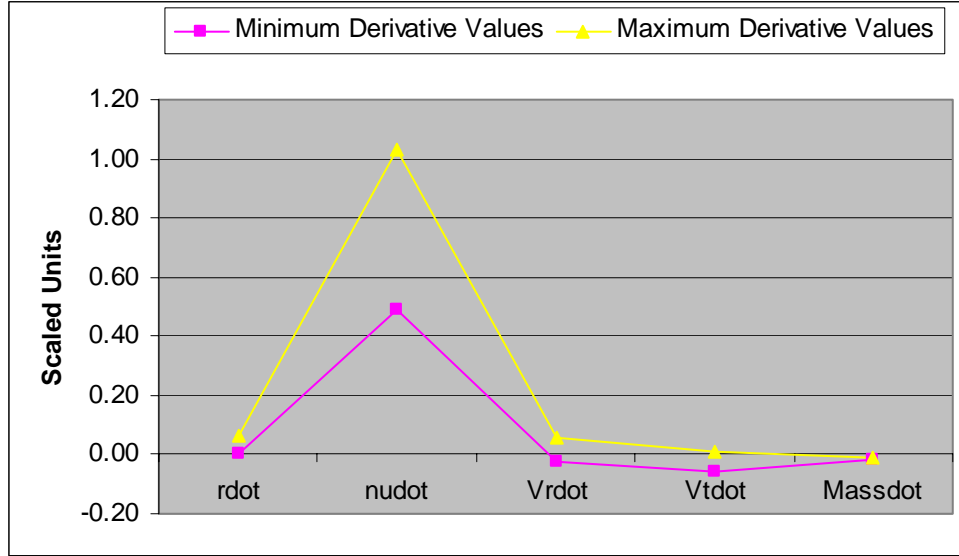
Figure 15   Actual Range of State Derivates

When these plots are examined the variable that needs the most attention is the range of minimum and maximum thrust for the first control variable. Here, the range is from 0.00 to a maximum scaled value of 0.0151. Even though this may seem like a small range in which to operate, it actually is a large improvement over the non-scaled that only varied from 0.00 to 0.000092 km-kg/s^2.[3]

To affect a better balance in the thrust force there are two methods. The easiest and least effective method is to recompute the scaling factors and redefined the non-dimensionalized units starting with dictating force units to be more advantageous, say 0.000075 vice 0.0075 as above. Then using this as one of the three fundamental units, define 2 more and let the other units be derived from solving equations much like Equations (44-48). Similarly, since the mass units was rather arbitrary and only affected the mass and force units, one could balance the well behaved mass state with the less well behaved thrust force, i.e. multiply the mass unit by an order of magnitude will increase the thrust force by an order of magnitude.

The much preferred and more thoughtful method is to insert an additional balancing factor in the dynamics, cost, path, and events where any force term is present.

---

[3] 92mN is the maximum thrust. However, since the distance units normally used for astrodynamics is kilometer vice meters, then using km as the distance unit to set up the problem, the thrust becomes 92x10^{-6} kg-km/s^2.

For example, such as in the previous scaling example using Equation (50), the Unscaled Dynamics method could be modified with:

$$\frac{d\bar{m}}{d\bar{t}} = \frac{U_{time}}{U_{mass}} \left(-\frac{\bar{\bar{T}}}{Isp * g_0} \right) T_{max} \qquad (54)$$

where

$$\bar{\bar{T}} \triangleq \frac{T}{T_{max}} \qquad (55)$$

Now, the normalized scaled thrust control, $\bar{\bar{T}}$, varies from 0 to 1 and the extra factor must be built into the equations to satisfy all the conversions. This same factor must be carried throughout all calculations. For example, if the performance function's purpose was to minimize ΔV, then one could use following cost function:

$$J = \int \frac{T}{m} dt \qquad (56)$$

If the balancing in Equation (55) is used, this must change to:

$$J = T_{max} \int \frac{\bar{\bar{T}}}{m} dt \qquad (57)$$

37

THIS PAGE INTENTIONALLY LEFT BLANK

III.   REACHABLE SETS

For the purpose of this thesis, a "reachable asteroid" shall be defined as one that can be visited by a spacecraft such that the spacecraft has enough fuel to return to Earth. Since this thesis concerns sample return missions, the visit is considered to be a rendezvous with a 90 to 120 day stay time to collect a sample. A "reachable set" (**A**) will be defined as the set of known asteroids that can be rendezvoused by a given spacecraft model and optimal use of the available fuel with the mission parameters.   By this definition it should be obvious that this set is dynamic given different spacecraft (fuel, mass, propulsion performance, etc.), mission requirements (return sample, number of asteroid visits, maximum flight time, etc.), and time.   This definition of reachable is generalized to not assume a specific launch window, which can change the available asteroids for a rendezvous and sample return mission. For instance, when the European Space Agency (ESA) Rosetta comet mission was postponed due the delay in the launch vehicle availability, the mission targets were changed since the launch window could no longer support the required trajectory. Basically, all asteroid targets take less or more fuel or time depending on the Earth's location relative to the target. Thus, the time of launch, relative to the synodic period between Earth and the target, changes which targets are within the reachable set for a given spacecraft and mission. When multiple asteroids are considered for rendezvous missions, a more simple analysis of synodic periods between two bodies is no longer period. This is why launch date is not specifically addressed in this topic. The reachable set would be a starting point to reduce the searchable targets for applicability if a specific launch window is required or desired. In other words, for this thesis a reachable set is only reachable in space and the smaller set of reachable targets in time and space is not considered.

On a two-dimensional representation, reachable sets will be plotted as in Figure 16 below. The horizontal axis represents the semi-major axis of the asteroid orbit about the sun and the vertical axis is the eccentricity. On a three-dimensional graph, inclination would be used to represent the orbit geometry in space.   The other classical orbital elements, such as true anomaly, right ascension of ascending node, and the argument of perigee, are not included for simplicity.
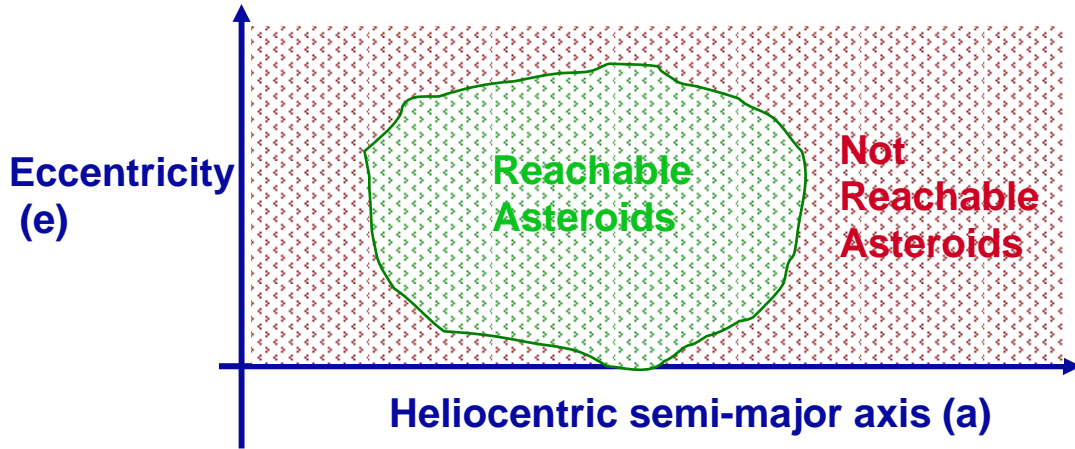
Figure 16   Reachable Asteroids

A.   ASTEROID TARGETS

As of July 2004, there are over 2800 Near Earth Asteroids (NEA), which are asteroids with perihelion distances less than 1.3 AU, out of a total database of over 250,000 asteroids in our solar system. For planning sample return missions, the NEA definition is arbitrary since it has no relation to a given spacecraft fuel, mass, and performance. However, physical properties are vitally important of the asteroid. For a spacecraft to successfully orbit and touch down on an asteroid, the asteroid must be of a sufficient size, density, and shape to have an acceptable gravity such that the spacecraft can enter a stable orbit around the asteroid. A good assumption on the necessary size is approximately 150 meters in diameter. Since asteroid size is estimated from the amount of light reflected from the surface and referred to as the Absolute Magnitude (*H*), there is a large variation in the possible size of the target depending on the assumed albedo. For instance, assuming an albedo range of 0.05 to 0.25, the asteroid size may range from 170 to 380 meters in diameter for an *H* of 21. Then using this absolute magnitude, the list of possible targets within our solar system large enough to consider a rendezvous mission reduces only by about 900. However, a vast majority of all these are in the either not within the inner solar system or far enough out that the spacecraft modeled has no chance to conduct a single sample return mission, much less have fuel for more than one asteroid visit.

Thus, to limit computational time a limit on semi-major axis, *a*, of the asteroid orbits will be placed at 2.1 AU. Based on Equations (20) and (22) we can compute a total

available ΔV from the nominal mission parameters and find that the maximum coplanar, circular rendezvous that can be achieved is just over 1.65 AU. Since this does not account for any fuel to return to Earth, the limit of only searching asteroids with $a < 2.1$ AU leave sufficient margin for any assumptions made. This limit, along with the absolute magnitude limit, reduces the possible targets that can be included in the reachable set to a maximum of 6157 targets, plotted in Figure 17. This thesis will try to deliver a method to further reduce this target set by another two orders of magnitude.
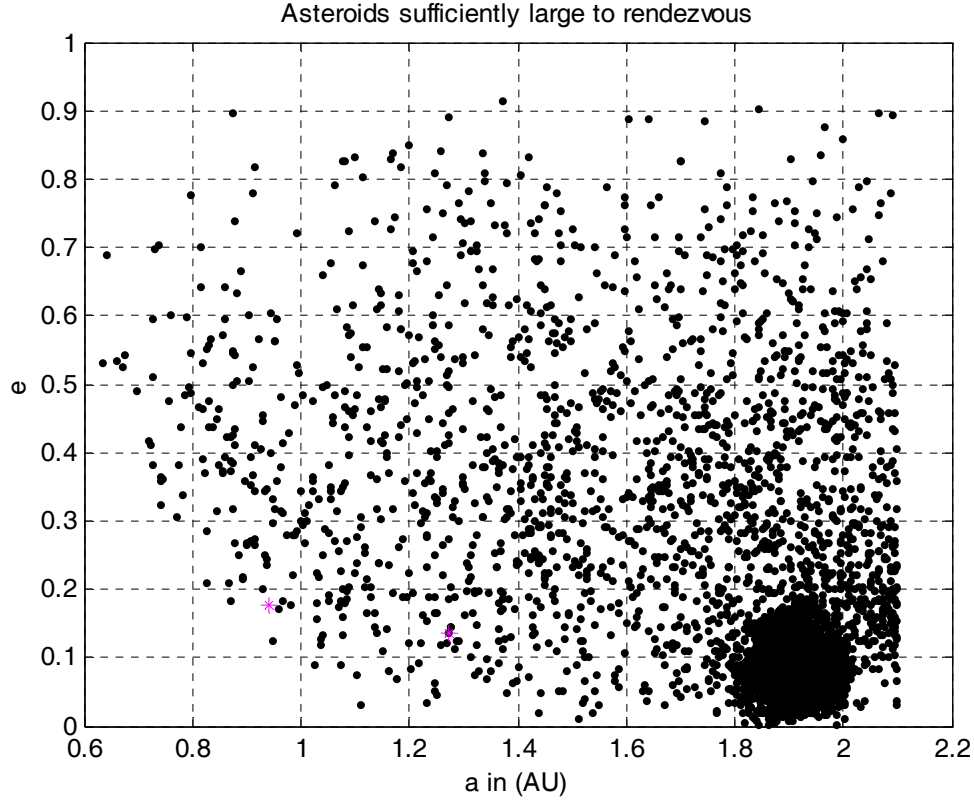
Figure 17   Inner Solar System Asteroid Orbits[4]

B.   CONTINUOUS BOUNDARY ISSUES

The two degree of freedom (DOF) reachable set projection on the a vs. e plot, such as in Figure 16 is a continuous region that surrounds a group of orbits generally characterized by their size and eccentricity. The boundary of this region is the limiting orbit that the given spacecraft can achieve. The discrete points on the plot where asteroid

---

[4] Note the dense cluster between 1.8 and 2.0 AU range. These asteroids are in the dense main belt of the Hungaria group and 90% have a high inclination between 16 and 35 degrees.

orbits are located will be in or out of this region. This region is found by computing families of optimal trajectories and weighting the importance in the semi-major axis and eccentricity in a cost function. By setting $F = 0$ in Equation , we have Mayer cost functions of:

$$\text{minimize } J_1 = \alpha_1 a + \beta_1 e \qquad \text{where: } \alpha_1 + \beta_1 = 1 \tag{58}$$

$$\text{minimize } J_2 = \alpha_2 a + \beta_2 e \qquad \text{where: } \alpha_2 + \beta_2 = 1 \tag{59}$$

As the two weights, α and β, are varied for each cost function, the boundary points on the reachable set found. Just as it takes an infinite number of points to fill in a continuous boundary, it would take an infinite number of optimal solutions, such as those when α and β weights are varied from 0 to 1, to exactly define the reachable set. Also, it would be contradictory to have a non-zero α or β term in each of the two performance index, since it makes no sense to try to maximize a or e at the same time as trying to minimize it. Thus, if you have a five increment linear sweep of α and β from 0 to 1 where $\alpha_{1,2} \text{ and } \beta_{1,2} \in \{0.0,\ 0.25,\ 0.5,\ 0.75,\ 1\}$ such that if $\alpha_1 > 0 \rightarrow \alpha_2 = 0$, $\alpha_2 > 0 \rightarrow \alpha_1 = 0$, and likewise for β, then it would take 20 optimal trajectories for every option to be computed. This situation was completed for the two DOF asteroid rendezvous without a sample return, and shown in Figure 18. Even though there is one non-validated point plotted here (max a where blue and red lines join), this is just used for illustrative purposes of real trajectories creating a reachable set with real asteroids.

This situation presented clearly demonstrates several problems with the generalized reachable set definition. First, even with 20 possible points (some lie on top of one another) there are large regions where the reachable set is ill defined due to the linear interpolation between the sparse data points. Also, notice the non-linear mapping between the trajectory solution points and the linear weights. Secondly, many possible target asteroids lie very near the reachable set boundary separating asteroids that have feasible trajectories to rendezvous and those that do not. There are several methods to approach these problems.
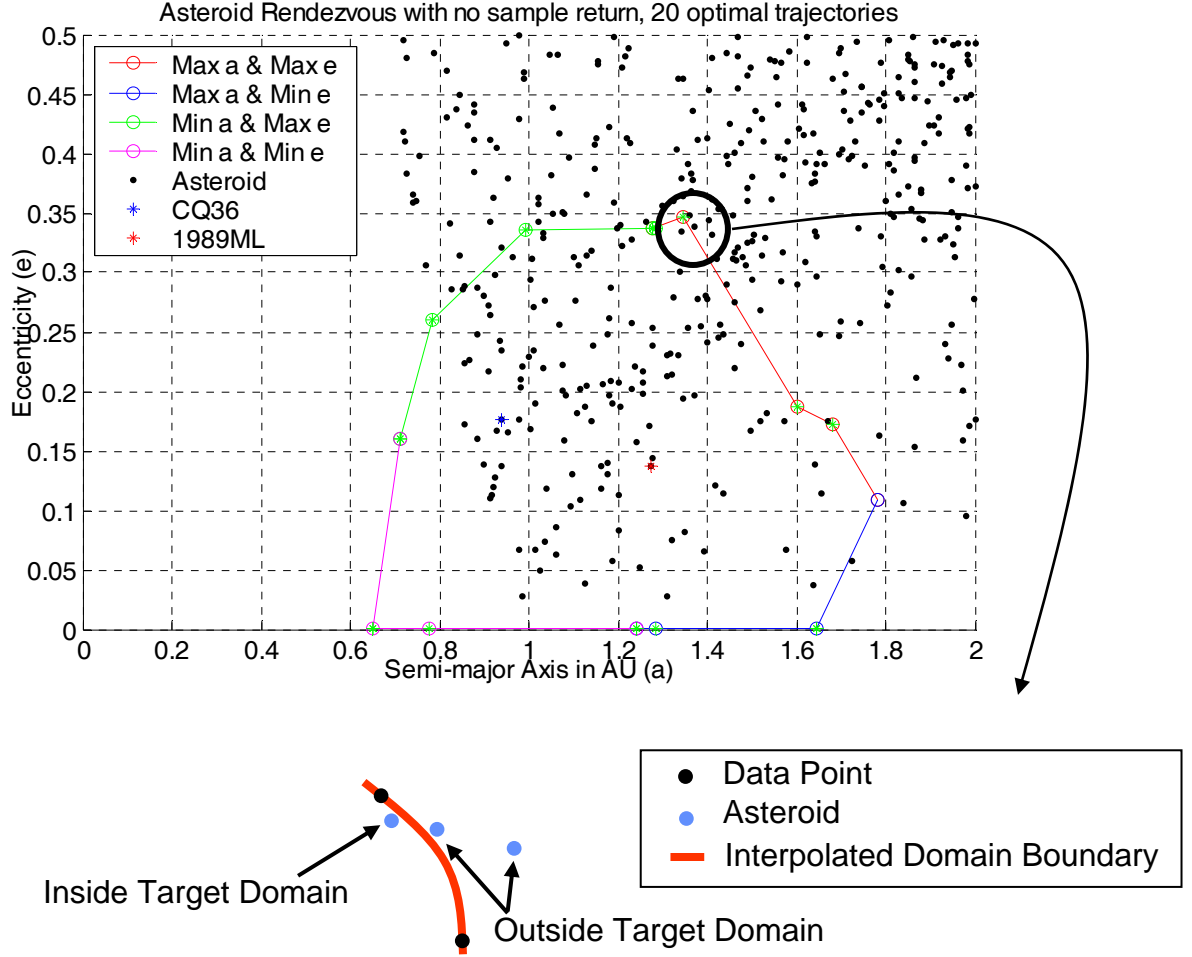
Figure 18   Reachable Set with 20 Trajectories and Many Ambiguities

After forming an initial reachable set, if there is an area of interest due to the desirability of a specific asteroid or just clarification on feasibility is required, the most simplistic way to achieve this is to add an additional data point where needed. Since a new weight on the performance index can not guarantee where the data point will show up due to the highly non-linear mapping, a new performance index could be created. In Figure 19 the data point required is identified and the semi-major axis where this would be beneficial is determined, for example the a_2 point. Now, the required semi-major axis is defined in the optimal control problem as an event (final $a = a_2$) and then the remaining parameter is maximized or minimized, $J_1 = e$ in this case. Additionally, another method would to be to check the feasibility of reaching an individual asteroid by change the cost function to minimize fuel and set asteroids a and e as events.

43
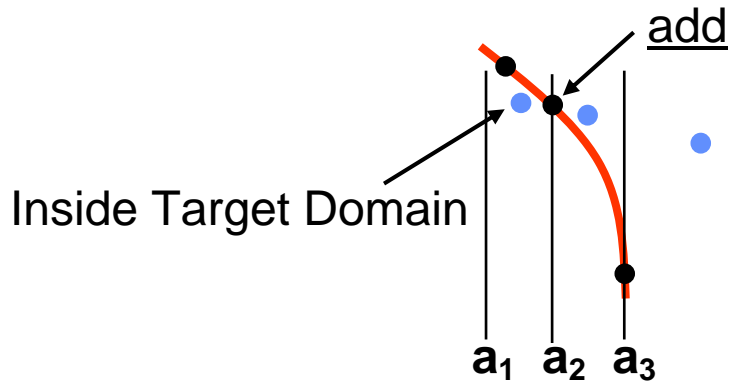
Figure 19   Adding Definition to Reachable Set Boundary

Recall that brute force methodologies were previously rejected when discussing multiple asteroid rendezvous and return missions since target sets in the thousands quickly turn into millions of possible combinations to consider. This is the very reason why limiting the target sets with a reachable set is necessary. Thus by proposing running optimal control problems on the order of 20 or more times, only to do more in-depth solutions to clarify those results is not very enlightening to a general methodology to achieve easier mission planning.

C.   INNER AND OUTER APPROXIMATIONS

Since brute force methods are not feasible or practical, the reachable set can better start to be definitized by first examining the four solutions when semi-major axis and eccentricity is fully maximized and minimized, or in other words, when $\alpha = 1$ or 0 and $\beta = 1$ or 0.   These four optimal control problem solutions can be visualized in Figure 20
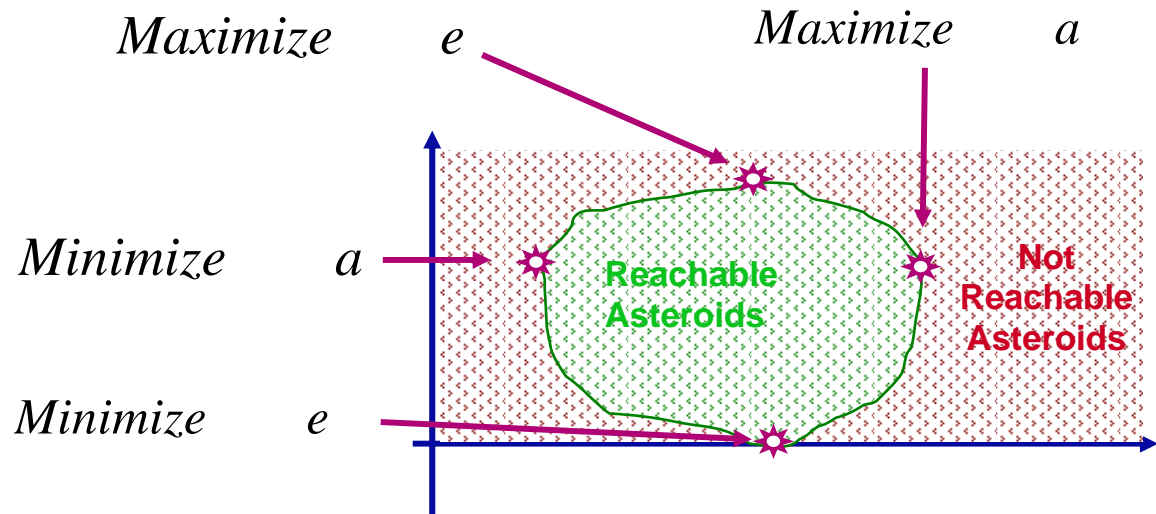
Figure 20   Four Extremal Points on Reachable Set

There will be two permutations within the reachable set definition that will be of benefit that will use the extremal points to further divide the reachable set. An "inner approximation" shall be defined as the reachable set by forming the convex polytope of the 4 extremal solutions of the performance index.
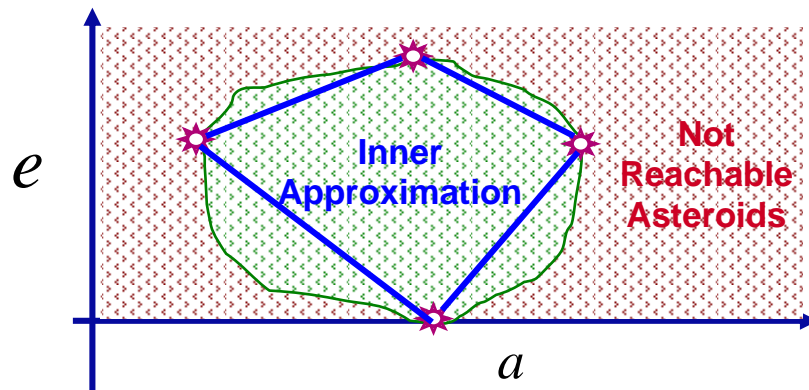
Figure 21   Inner Approximation

An example of an inner approximation is shown in Figure 21. This is useful since with only four optimal control problem solutions, a large part of the mission achievable asteroid targets can be defined. These inner approximation targets have the highest confidence of not laying so close to a boundary that there is little margin left in the fuel budget to make them risk acceptable targets. However, even more useful is the concept of the outer approximation. The outer approximation is defined as of the reachable set

45

created by excluding any asteroid with $a < a_{min}, a > a_{max}$, and likewise for e. This outer approximation mathematically excludes the asteroids that can absolutely not be reached, since they lay outside the extremal points on any reachable set. The asteroids left inside the outer approximation reduces the possible targets so that detailed planning, such as examining the ignored orbital elements of the asteroids, can be done with as much efficiency as possible.
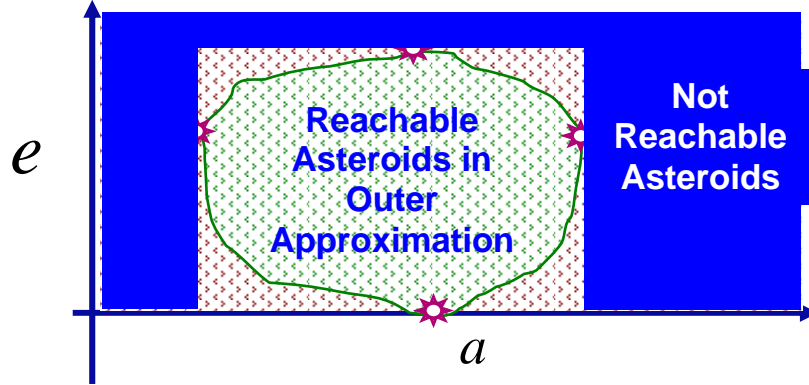
Figure 22   Outer Approximation

A more real example is shown how this would affect the previous non-return mission discussed. Figure 23 demonstrates with just four optimal control problem solutions, a majority of the possible asteroids targets that our outside the outer approximation can be simply excluded from further study with higher fidelity models. Also, the previous examples of how to better delineate between specific boundary locations can still be conducted if required.
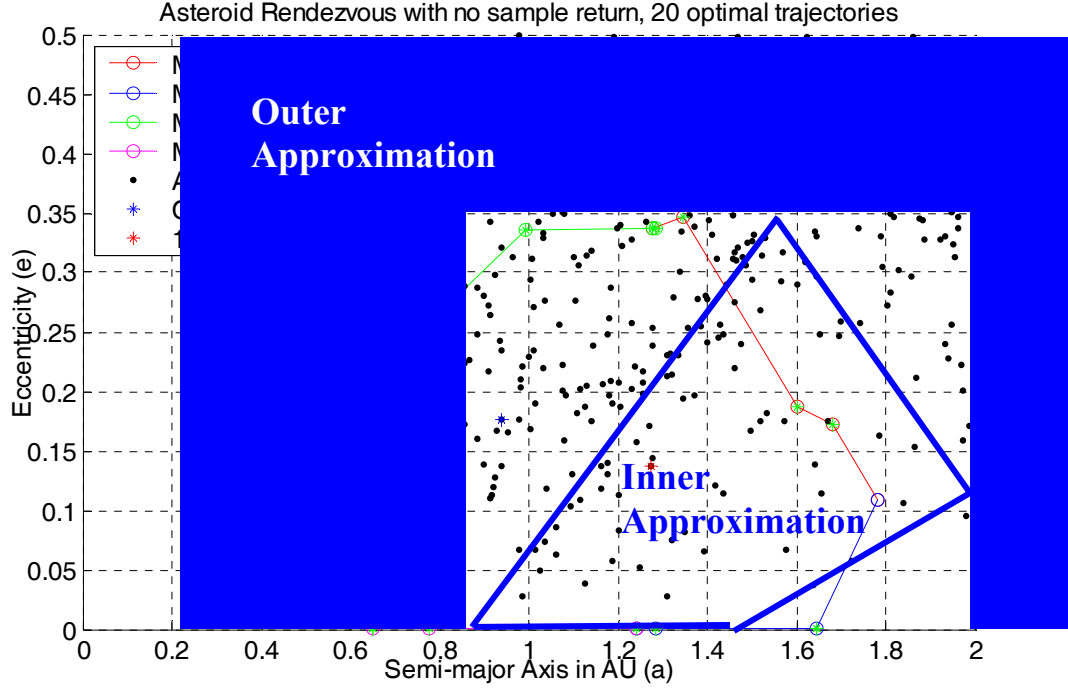
Figure 23   Inner and Outer Approximation Example

So far, the relative importance of an inner and outer approximation has yet to be fully described. It may seem that the reachable set definition in Figure 18 is far superior that that in Figure 23, especially since the 20-point reachable set boundary is more distinct than the four-point boundary that defines the inner and outer approximations. However, when a three-dimensional reachable set is required, the dynamics and performance index must be modified to take inclination into account. A new performance index would be:

$$minimize\ J_1 = \alpha_1 a + \beta_1 e + \gamma_1 i \qquad \forall\ \alpha_1 + \beta_1 + \gamma_1 = 1 \qquad (60)$$

$$minimize\ J_2 = \alpha_2 a + \beta_2 e + \gamma_2 i \qquad \forall\ \alpha_2 + \beta_2 + \gamma_2 = 1 \qquad (61)$$

Now, to get the same relative number of boundary points (five increments of the new weight, γ, and include inclination of targets it would take 100 simulations that would need to be run and validated. So for a number of incremental values in the weights, referred to as N_I, then the number of required non-linear optimization problem (NLP) solutions required for a single sample return mission is [Ref. 11]:

47

$$NLP_{\substack{single \\ sample}} = N_I^3 - N_I^2 \tag{62}$$

However, to solve for the inner and outer approximations, only 2 more NLPs must be solved, to find the extremal cases maximum and minimum inclination[5]. Adding more dimensions (DOF), such as any of the other classical orbital elements, will compound the required effort of brute force methodologies when creating reachable sets and a generalized equation would be:

$$NLP_{\substack{single \\ sample}} = N_I^{DOF} - N_I^{(DOF-1)} \tag{63}$$

In contrast, creating inner and outer approximations only requires two times the degrees of freedom (2 x DOF) number of solutions. Since defining a reachable set is a mathematically valid method to decrease the number of possible target opportunities by at least an order of magnitude to aid in further detailed mission planning, this resource saving process becomes much more important when three or more degrees of freedom (performance index items) are required.

---

[5] In single sample return mission if it can be assumed that the spacecraft originates from Earth (i=0), then the cases to minimize inclination are not necessary. This assumption will not be valid for any multiple sample return, since the return flyby will rarely have a zero inclination.

IV. ASTEROID RENDEZVOUS

This thesis will explore a methodology to determine the set of reachable targets for a multiple asteroid rendezvous and sample return mission. To reduce the overall risk in the mission, JPL has decided that each sample should be returned to Earth prior to rendezvousing with the next target. The next several chapters will build up to the overall problem with initially more emphasis on the optimization and then in later cases more emphasis on interpreting the results. Figure 24 shows a general representation of a low thrust, multiple-revolution Earth to asteroid rendezvous geometry. The spacecraft, launched out of Earth's sphere of influence (SOI) by a launch vehicle, departs Earth and will rendezvous with an asteroid. This case is formulated in only two dimensions to simplify the solution and analysis. Because "hard" knots [Ref. 7] are not needed, this case has the most tools available to prove that the necessary conditions for optimality are met. As mentioned previously, in this and all cases the objective function is to maximize or minimize a convex combination of semi-major axis and eccentricity. This mission with just one "leg" or point-to-point trajectory is very simplistic, but very illustrative to how the problems are set up, solved, and validated.
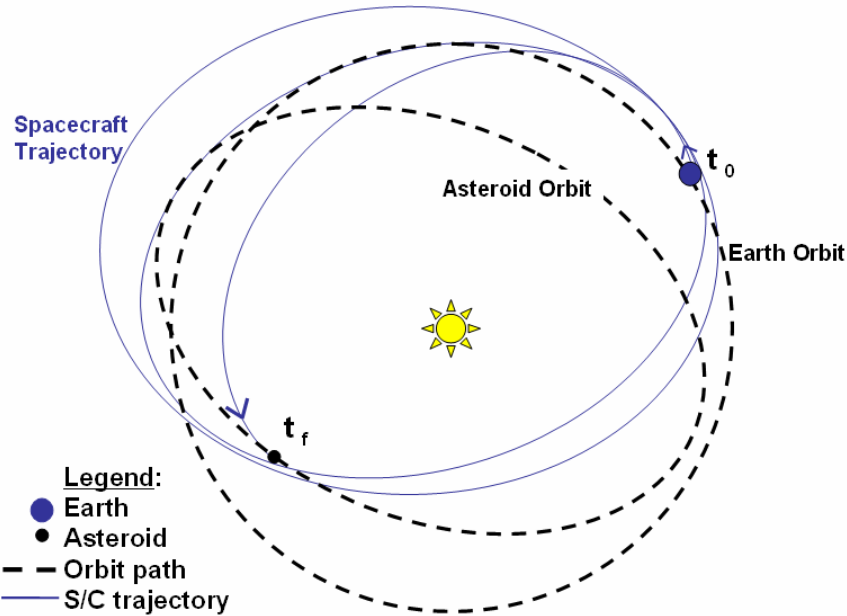

Figure 24 Earth to Asteroid Low Thrust Trajectory Representation

The simplest test case to start would be to first constrain the rendezvous to a circular orbit. Thus, the transfer is just an optimal circular to circular low thrust trajectory and maximizes the final orbit radius (same as maximizing semi-major axis with zero eccentricity). Additionally, not taking into account the thrusting constraints of a SEP motor nor with an initial earth escape velocity boost ($C_3 = 0$) will allow the results to be compared to known analytical approximations of optimal trajectories and remove the LaGrange multipliers associated with those path constraints.

A. DYNAMICS, COST, EVENTS, AND PATH FORMULATION

The dynamics for a circular or elliptical formulation is exactly the same as Equations (1-5). The general case for the cost function, which is convex combination of the semi-major axis and the eccentricity of,

$$J = \alpha a_f + \beta e_f \qquad \text{where: } \alpha + \beta = 1 \tag{64}$$

for the circular case becomes:

$$J = r_f \tag{65}$$

The event constraints, which will ensure the problem start and end points for a state or time are either met or optimized was in a general form in Equation (17) and is now specifically set to (normalized values):

$$\mathbf{e}_l = \mathbf{e}_u = \begin{bmatrix} r_0 \\ \nu_0 \\ v_{r_0} \\ v_{t_0} \\ m_0 \\ r_f \\ \nu_f \\ v_{r_f} \\ v_{t_f} \\ m_f \end{bmatrix} = \begin{bmatrix} 1 \\ free \\ 0 \\ 1 \\ M_{wet} \\ free \\ free \\ 0 \\ \sqrt{1/r_f} \\ M_{dry} \end{bmatrix} \tag{66}$$

These equality constraints have the same upper and lower constraint. The "free" states are those that have no constraints and thus the optimization routine is free to let those start and end on any convenient value. Since the initial and final orbit is circular, there is

50

no radial component of velocity, v_r, to start or end. The transverse velocity magnitude, v_t, is initialized on the same value as Earth and dependant on the final radius, which is a dependant variable in the cost function. The mass state is set to the wet mass of the S/C and all the fuel will be used when the final mass state is equal to the dry mass of the S/C. The only path constraint is on the thrust magnitude, as previously stated. This ensures that no time during the trajectory the maximum thrust exceeds the engine's limits, or $T_{mag} < 92 mN$.

B.   BOUNDS, GUESS, AND NODES FOR PROBLEM FORMULATION

All states, controls, and time bounds are specified in the problem initial setup; however these can differ from the constraints mentioned in the previous paragraph. If a constraint is placed at a specific point (start, end, or interior) then it would be considered an event constraint and if the constraint is during the entire time span of the problem, then it is considered a path constraint. The event, path, and dynamical constraints can be complex and non-linear functions where the actual constraint is not specifically known at the problem start. For instance, since in SEP trajectories the final maximum thrust magnitude is not known until the final maximum radial distance from the sun is computed, this constraint must be computed within the optimization. In DIDO [Ref. 7], "bounds" are specified at the start and used to either simply constrain a state, control, or time at an *a priori* value or more commonly to avoid values that would lead to singularities in the NLP. For definition purposes, a state, control, or time bound used to constrain a solution is called a "box constraint" and a bound used in limit the NLP solution algorithms will just be called a bound and will be assumed to not be any determination to the final solution. For the simple circular, low thrust, max radius problem the following bounds were used:

Bounded Variable	Lower Bound	Upper Bound	Note
Radius, r	0.5 AU	10 AU	Avoids singularity at 0 AU
Transfer Angle, θ	-10π	10π	5 revolutions
Radial Velocity, v_r	$-10*|V_{earth}|$	$10*|V_{earth}|$	
Transverse Velocity, v_t	$-10*|V_{earth}|$	$10*|V_{earth}|$	
Mass, m	$0.1*M_{wet}$	$10*M_{wet}$	In reality $M_{dry} \cong 0.8*M_{wet}$
Thrust Magnitude, T	$-1000*T_{max}$	$1000*T_{max}$	
Thrust Direction, θ	$-\pi$	π	Covers 360° of Thrust
Time, t	0 years	10 years	

Table 2.　　State, Control, and time Bounds

The bounds in Table 2 are actually used in every optimal control problem solution presented in this thesis. Since there is no constraining element in using these bounds, they seem to provide adequate limits on values to speed computation without limiting the search space for real state, control, or time values. The thrust direction may seem like the most restrictive bound and it does control the resulting thrust direction output of the optimization routines, but only so that it does not need to be post-processed to display the thrust direction in a similar range (i.e. converting 4π thrust direction into 0).

The nodes in the NLP are discrete Legendre-Gauss-Lobatto (LGL) points where the solution to the problem is calculated. Since this problem already has five states and two controls defined, the number of parameters to solve are these 7 variables times the number of nodes plus the initial and final time. Thus, a 20 node solution might not have much fidelity but only needs to solve 142 parameters simultaneously. However, an 80 node solution has 562 parameters to solve. Most solutions will have at least 60 and up to 200 nodes in the final solution so the circular orbits look smooth and to ensure there is enough information to make assessments.

Guesses must be provided for every at every interior, start, or end point that can be used to define an event. As previous mentioned, the "no guess" type of initialization

for a 20 node solution was performed and automatically those results (optimal or not) were directly used as the guesses for a higher node solution (typically 80). This method increased the computational speed of just performing an 80 node optimization with a low fidelity guess and the initial guess never had to be re-evaluated. Thus, using 16 guesses, the 142 parameters are solved for the 20 node solution, all of which are used as the guesses to generate the 562 parameters in the 80 node solution. The next table presents the guess used for all optimization problems with no return to Earth, and thus no interior point or knot to consider. The actual start and end is just one typical example and since most of the guesses match the event constraints, the start and end points are very simple.

Variable	Start Guess	End Guess	Actual Start	Actual End
Radius, r	1 AU	2 AU	1 AU	1.65 AU
Transfer Angle, θ	0	3π	0	2.71π
Radial Velocity, v_r	0	0	0	0
Transverse Velocity, v_t	$|V_{earth}|$	$0.707*|V_{earth}|$	$|V_{earth}|$	$0.779*|V_{earth}|$
Mass, m	M_{wet}	M_{dry}	M_{wet}	M_{dry}
Thrust Magnitude, T	0	T_{max}	T_{max}	0
Thrust Direction, θ	0	2π	-0.23π	-0.06π
Time, t	0 years	2.5 years	0 years	4.04 years

Table 3.   Guess for Problems without Interior Knots

C.   CIRCULAR, LOW-THRUST EARTH TO ASTEROID RENDEZVOUS SOLUTION

An illustrative solution for the simplest case follows and is described in Figure 25 - Figure 27. Since the circles (DIDO solution) are very close to the independently propagated states using the optimal control history, u(t)*, this shows a dynamically feasible result. Additionally, by integrating the acceleration during the trajectory, as in Equation (42), the total ΔV used was 6.58 km/s. This favorably compares to the ideal rocket Equation estimation of 6.59 km/s. Lastly, no boundary constraints, event

constraints, or bounds where broken for the DIDO solution or propagated trajectory of the optimal controls. Since the total error norm is low and the good agreement between the DIDO solution and control history propagation, this represents a feasible result.
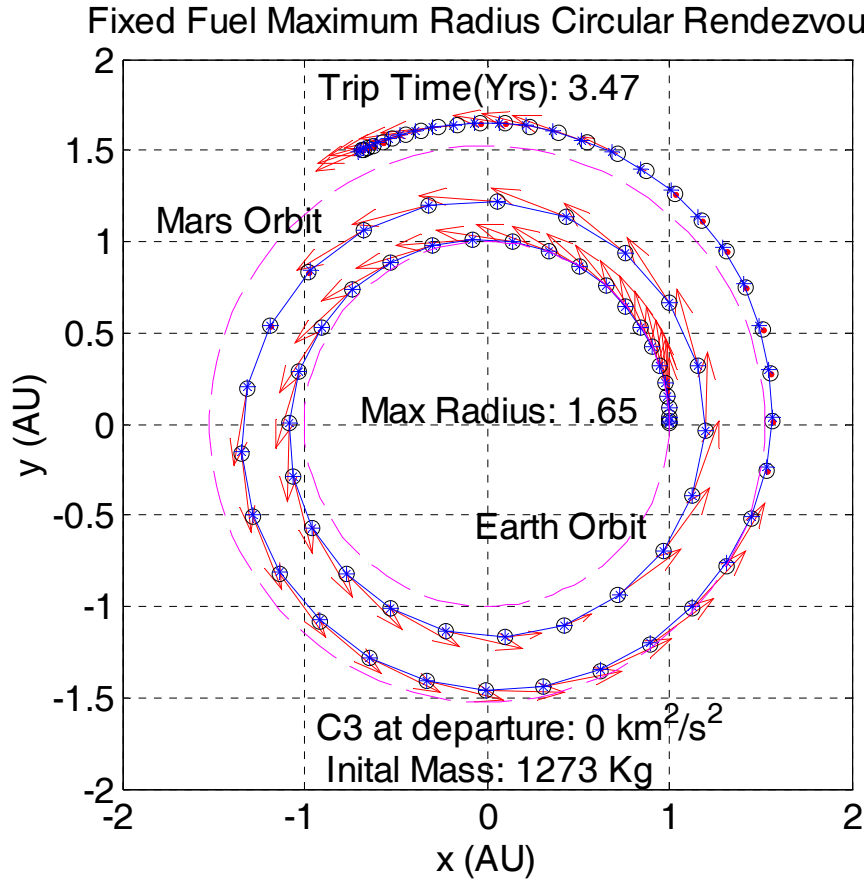

Figure 25   Circular Orbit Rendezvous (Result 1)[6]

Figure 25 shows the location of the spacecraft and the direction of the thrusting (if it is thrusting. Figure 26 shows the same information to include mass and velocity states to show a good correlation between the DIDO states (open geometric shapes) and the lines (propagated solution that has many more times the number of points).

---

[6] On a Windows XP based Pentium 4 CPU running at 1.3MHz and with 512MB of RAM, this problem took 117.7 seconds. On a Windows 2000 based Pentium M CPU running at 1.5MHz and 256MB of RAM the exact same problem took only 44.0 seconds.
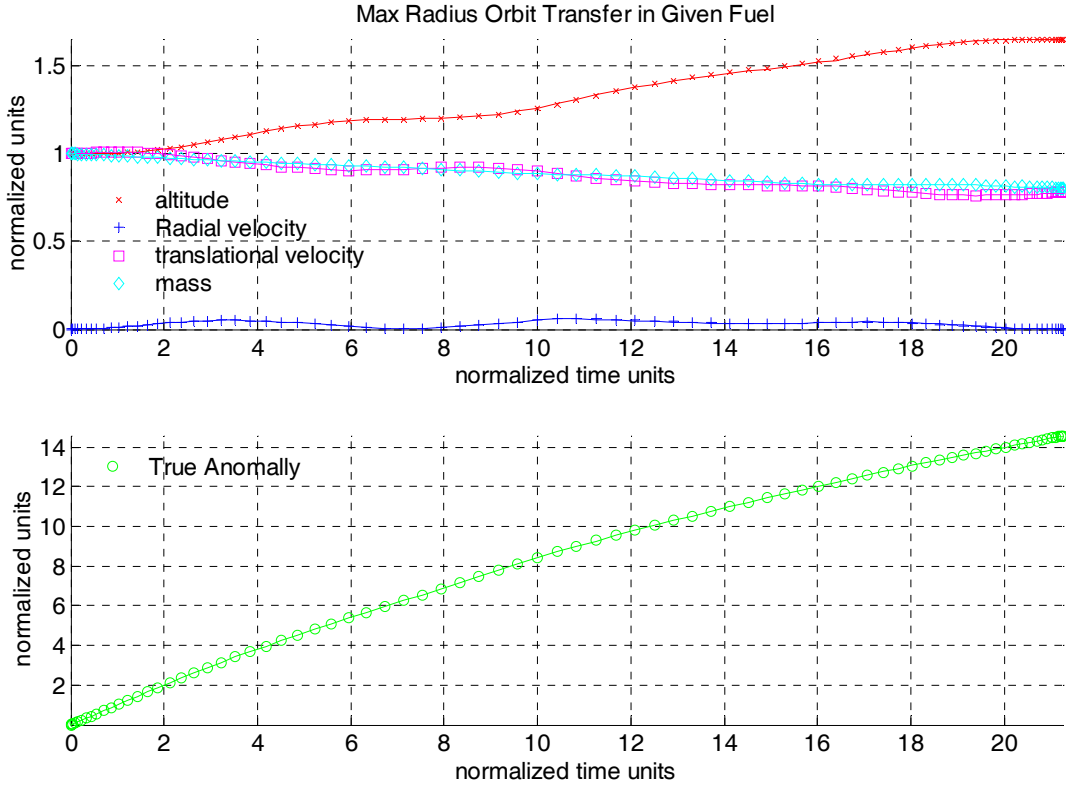
Figure 26   State History for Circular Rendezvous (Result 1)

The final orbit achieved by using all the fuel available is a circular rendezvous orbit at 1.65 AU. To check for optimality, first this result will be compared with known solutions. If this was an ideal impulse mission, the optimal coplanar, circle-to-circle orbit transfer would use a Hohmann Transfer. Using the Equation (38), to get to a 1.65 AU orbit from 1 AU would require an ΔV of 6.50 km/s. This is only 1.6% less than the total ΔV used by the low thrust engine which is less efficient due to off-axial thrusting and thrusting an non apoapsis locations. Secondly, this can be more accurately compared to an ideal circle-to-circle continuous low-thrust transfer using Edelbaum's solution, shown in Equations (35-37). The Edelbaum equations requires an ΔV of 6.59 km/s, or 0.2% more than DIDO's optimal result.
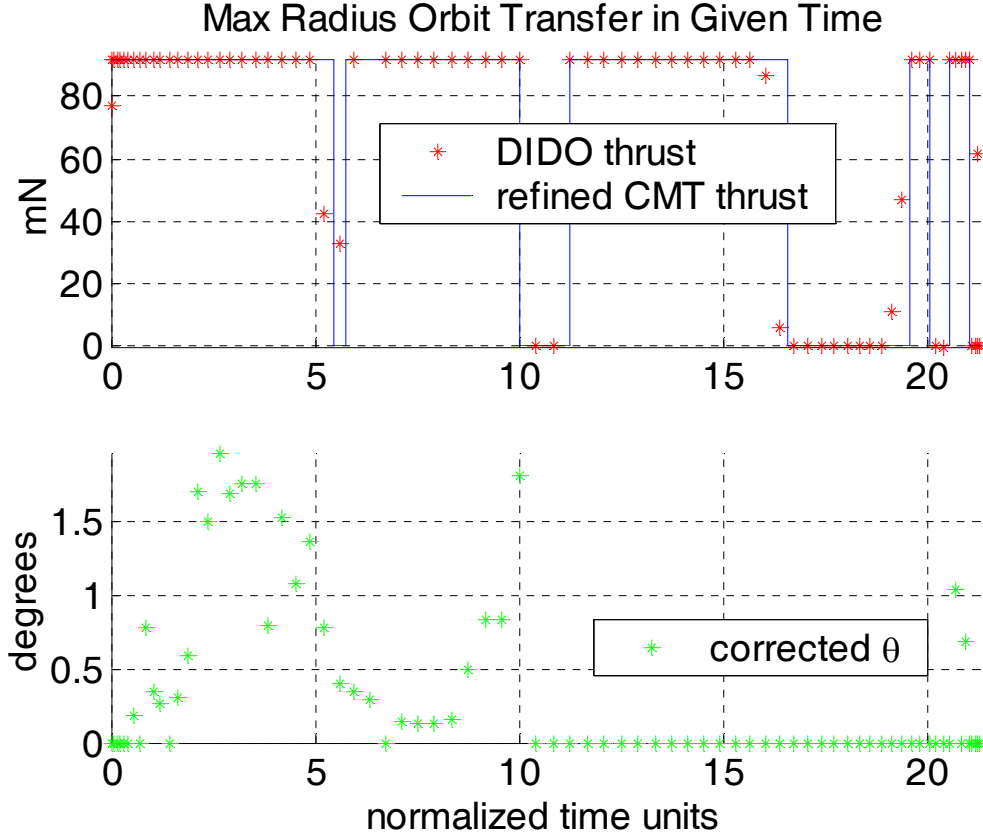
Figure 27   Circular Rendezvous Control History (Result 1)

The top portion of Figure 27 shows the DIDO thrust compared with the Covector Mapping Theorem controls. The bottom portion shows the thrust angle however the raw results are "corrected" to make plots easier to read in case the result was 360 degrees, vice zero, or the thurst angle is zeroed out if no thrusting occurs. When no thrusting occurs the angle can be random since it has no effect on the dynamics, cost, or other states. Overall, this is a good correlation between the theory and the result. Also as previously noted, the results must satisfy the Karush-Kuhn-Tucker (KKT) theorem conditions. From these conditions, a Switching function (S_T) on the control variables can be created and as such Equation (31) is restated below with specific controls substituted in:

$$S(\tau) = \begin{cases} \leq 0 & T(\tau) = T_{max} \\ \geq 0 \quad if & T(\tau) = 0 \\ = 0 & 0 < T(\tau) < T_{max} \end{cases} \tag{67}$$

$$S(t) = \lambda_{v_r} \frac{\sin\theta}{m} + \lambda_{v_t} \frac{\cos\theta}{m} - \frac{\lambda_m}{Isp * g_0} \qquad (68)$$

A plot of the switching function computed from DIDOs results of the costate dynamics is in Figure 28. This shows agreement between KKT conditions and the thrusting and coasting phases. Additionally, the points where the switching function is near zero is where the thrust is throttled and neither full thrusting or off is the optimal condition.
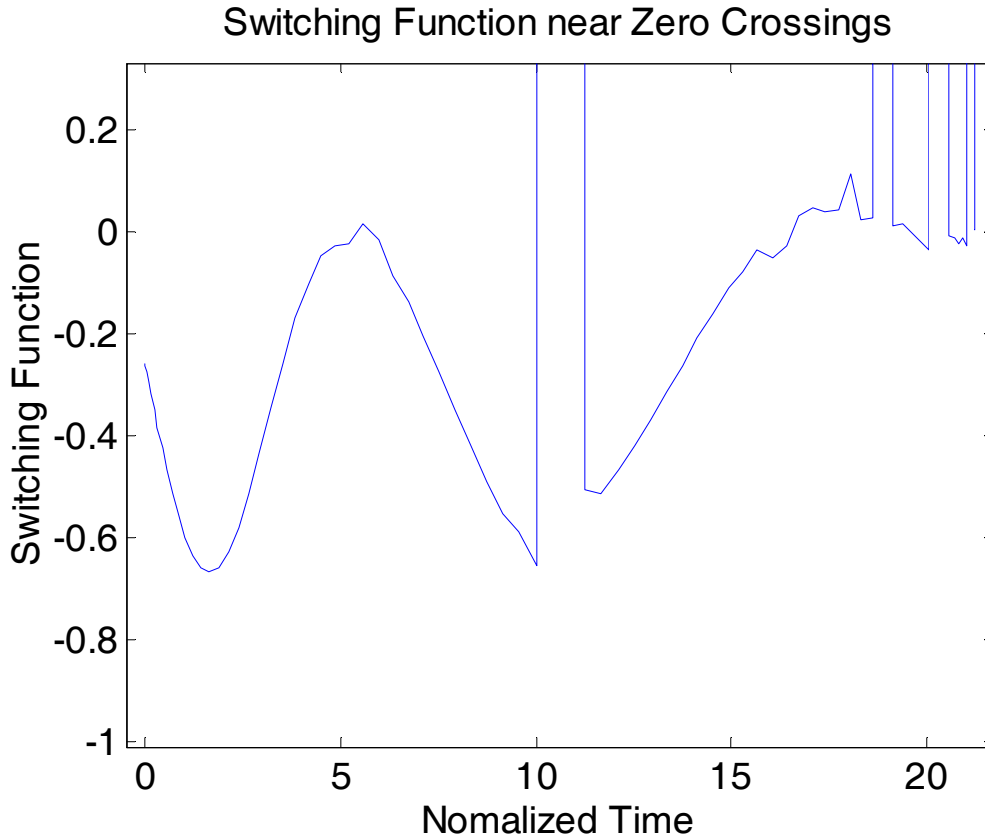

Figure 28   Switching Function

Equation (34) shows that the Hamiltonian should equal zero for all time when time is not explicit in the path constraint or the dynamics. This Hamiltonian is shown in Figure 29 with a mean value of -0.00072294. This is a good result and indicates that the solution is optimal. Also, as previously stated, Equation (26) is the second order condition for optimality and requires that the Hessian of the Hamiltonian is positive semi-definite. This also holds true for an augmented Hamiltonian.

$$\bar{H}(\lambda, x, u, t) = F + \boldsymbol{\lambda}^T \mathbf{f} + \boldsymbol{\mu}^T \mathbf{h} \tag{69}$$

$$\bar{H} = \lambda_r v_r + \lambda_\upsilon \frac{v_t}{r} + \lambda_{v_r}\left(\frac{v_t^2}{r} - \frac{\mu}{r^2} + \frac{T}{m}\sin\theta\right) + \lambda_{v_t}\left(\frac{-v_r v_t}{r} + \frac{T}{m}\cos\theta\right)\ldots$$
$$+ \lambda_m\left(-\frac{T}{Isp * g_0}\right) + \mu_{T_{\max}} T \tag{70}$$

$$\frac{\partial \bar{H}}{\partial T} = \frac{\lambda_{v_r}}{m}\sin\theta + \frac{\lambda_{v_t}}{m}\cos\theta - \frac{\lambda_m}{Isp * g_0} + \mu_{T_{\max}} \tag{71}$$

$$\frac{\partial^2 \bar{H}}{\partial T^2} = 0 \geq 0 \quad \text{(trivial case, always true)} \tag{72}$$

$$\frac{\partial \bar{H}}{\partial \theta} = \lambda_{v_r}\left(\frac{T}{m}\cos\theta\right) - \lambda_{v_t}\left(\frac{T}{m}\sin\theta\right) \tag{73}$$

$$\frac{\partial^2 \bar{H}}{\partial \theta^2} = -\frac{\lambda_{v_r} T}{m}\sin\theta - \frac{\lambda_{v_t} T}{m}\cos\theta \geq 0 \tag{74}$$
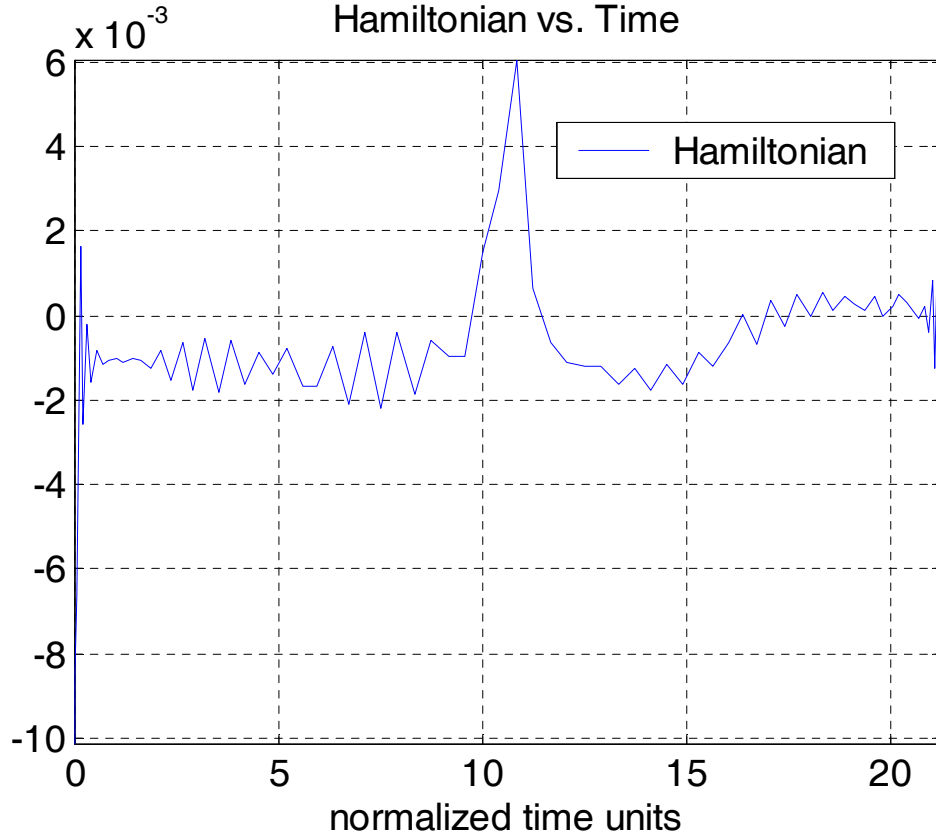

Figure 29   Hamiltonian for Circular Rendezvous (Result 1)

58

Figure 29 demonstrates near zero conditions and Figure 30 demonstrates that the second partial derivate of the Hamiltonian with respect to the controls are positive semi-definite. Both are required conditions by Pontryagin's Minimum Principle are indeed satisfied.

Thus, this solution (Result 1) for the circular, 2-D, low thrust transfer without SEP constraints has been shown to be feasible, satisfying the necessary conditions for local optimality, and also compare very favorably with known theoretical solutions. For future results, not all of these comparisons can be made or will be explicitly shown. For elliptical orbits there are no low-thrust analytical solutions to compare results with. For trajectories that must rendezvous for a period of time with an asteroid, the solution requires the use of a soft or hard knot in the solution [Ref. 7] and will not have dual variables to construct Hamiltonians, CMT controls, or switching functions. However, DIDO does state at the end of an optimization routine if it believes the solution is locally optimal.
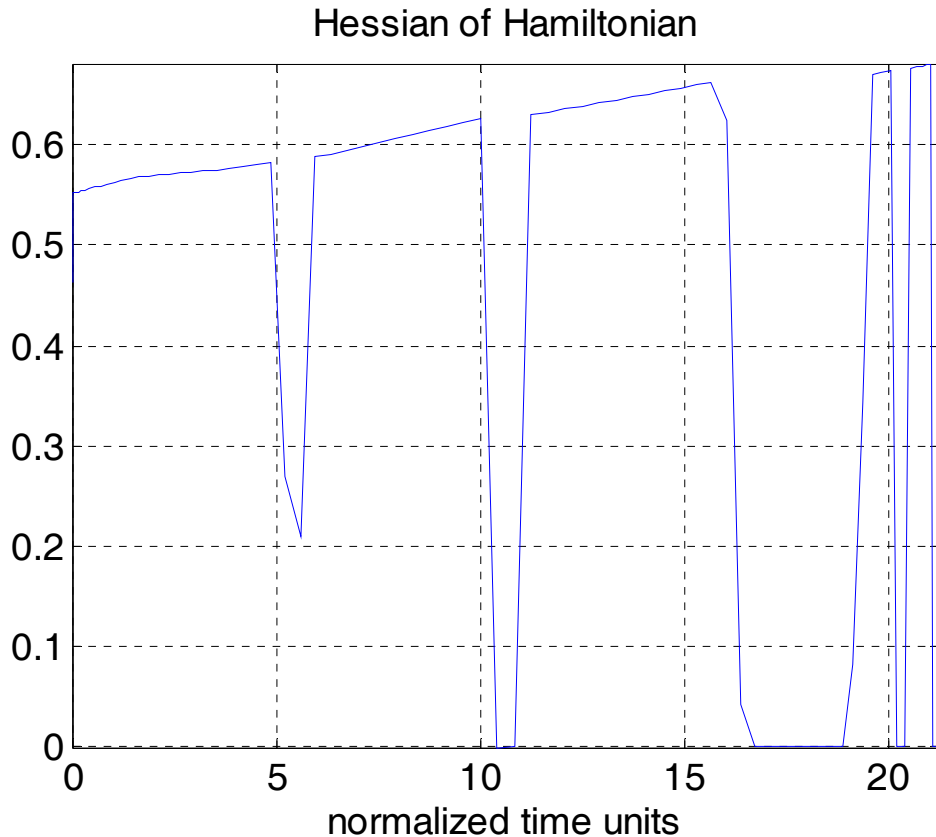

Figure 30   Hessian of Hamiltonian (positive semi-definite)

D.   ELLIPTICAL, LOW-THRUST EARTH TO ASTEROID RENDEZVOUS SOLUTIONS

This optimization run compute trajectories for Earth to asteroid rendezvous' that can now be elliptical, add a C_3 boost optimization routine, and add a limited SEP propulsion model of the NSTAR engine. This is limited in that the I_{sp} is not adjusted down from the maximum as in a real model and it is possible to have thrusts below 19 mN (though unlikely). The dynamics remains the same as Equations (1-5). The cost function, which is convex combination of the semi-major axis and the eccentricity written as,

$$J = \alpha a_f + \beta e_f \qquad \text{where: } \alpha + \beta = 1 \tag{75}$$

and can be weighted to maximize or minimize either a, e, or a combination. The event constraints, which will ensure the problem start and end points for a state or time are either met or optimized was in a general form in Equation (17) and is now specifically set to a range of inequalities (normalized values in the with the lower bound to the left and upper bound to the right in the set). Any non-constrained boundary events are considered free and the initial mass state is set dependant on a function of the C_3 boost optimized for launch.

$$\mathbf{e}_{[l,u]} = \begin{bmatrix} r_0 \\ \theta_0 \\ m_0 \\ \left(v_{t_0}-1\right)^2 + v_{r_0}^2 - C_3 \\ e_f \\ m_f \end{bmatrix} = \begin{bmatrix} [1,1] \\ [0,0] \\ f(C_3) \\ [0,0] \\ [0,1] \\ [M_{dry,}M_{dry,}] \end{bmatrix} \tag{76}$$

There are now two path constraints on the thrust magnitude. The first is to limit thrust to only positive values or zero and the second is to limit the thrust dependant on the NSTAR engine model maximum thrust available and is a function of range to the sun and time. Thus, two path constraint LaGrange multipliers are required to construct the Hamiltonian and for the Thrust switching function, which now becomes:

$$S(t) = \lambda_{v_r} \frac{\sin\theta}{m} + \lambda_{v_t} \frac{\cos\theta}{m} - \lambda_m \frac{1}{Isp * g_0} + \mu_{T_{max}} T(r,t,P_e) + \mu_{T_{min}} T(r,t,P_e) \tag{77}$$

The thrust (T) is computed in Equation (12) for a given range, time, and electric power available. Lastly, all the same bounds, guesses, and number of nodes were left unchanged from the previous problem formulation.

A DIDO result using equal weights (of -0.5) to maximize both *a* and *e* is shown in Figure 31, Figure 32, Figure 33, Figure 34, and Figure 35. This shows a trajectory that uses a large boost (C_3 equal to 7.75 km^2/s^2) to start the journey and uses the remaining 73 kg of fuel to complete the orbit transfer. The final result was *a* = 1.49 AU, *e* = 0.271, and a maximum orbit radius of 1.90 AU. A summary is shown in Figure 31 with and well correlated states are shown in Figure 32 and Figure 33. The color code in Figure 31 has green stars when the S/C is on the apoapsis side of its instantaneous orbit and blue on the periapsis side. Since it is more beneficial to thrust on the green side, one would expect to see more thrust vectors there. This result was deemed "locally optimal" by DIDO and the solution propagates well (using ODE113).
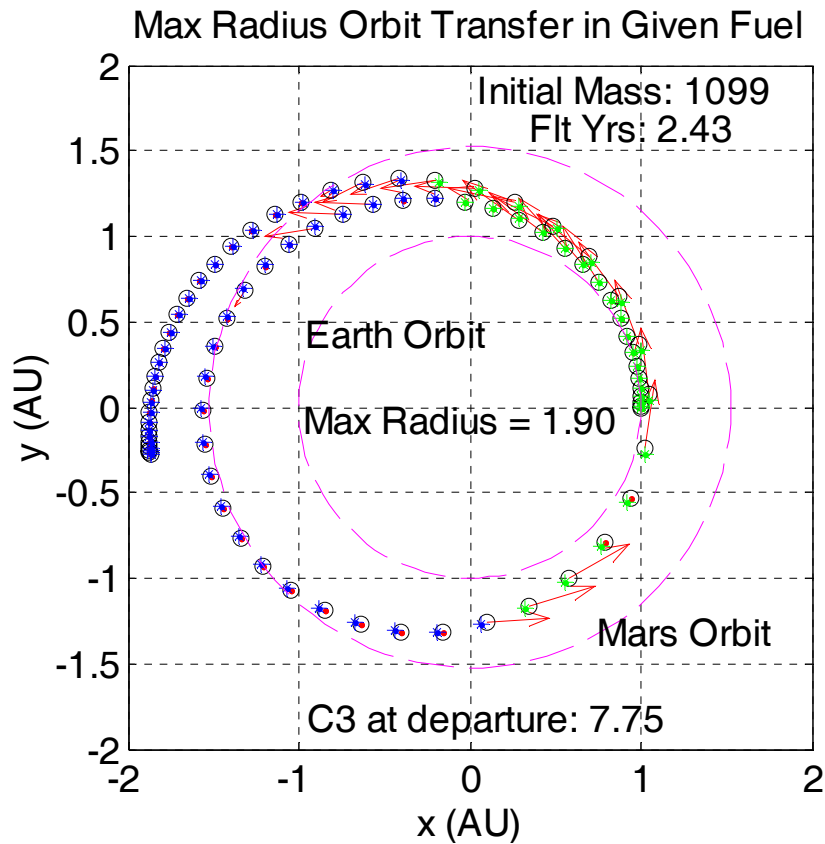

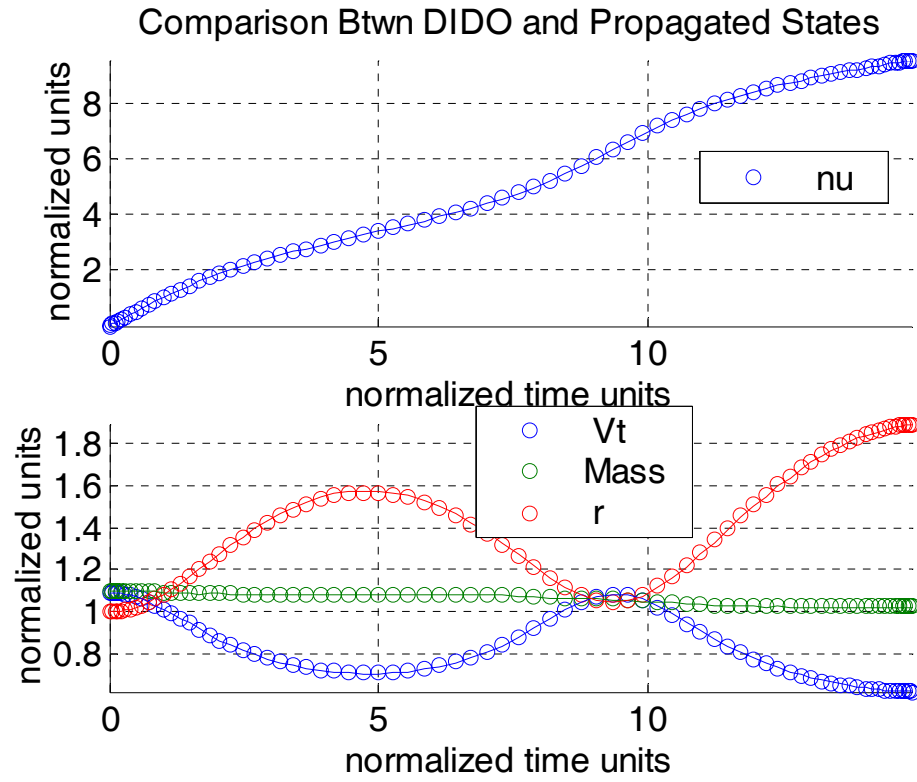Figure 31   Elliptical Rendezvous Orbit (Result 2)

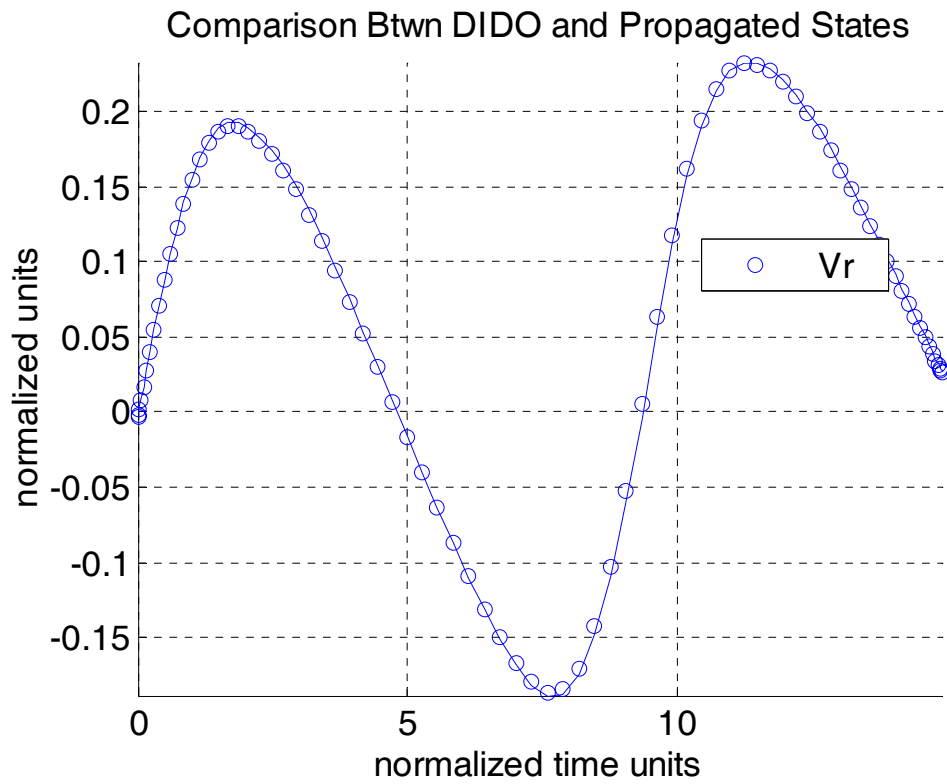Figure 32   Elliptical Rendezvous States (Result 2)

Figure 33   Elliptical Rendezvous States (cont) (Result 2)

The controls shown in Figure 34 closely follows the theoretically optimal CMT thrusting shown and the Hamiltonian in Figure 35 is fairly flat with a mean value of -0.0973. These are good indications of optimality.
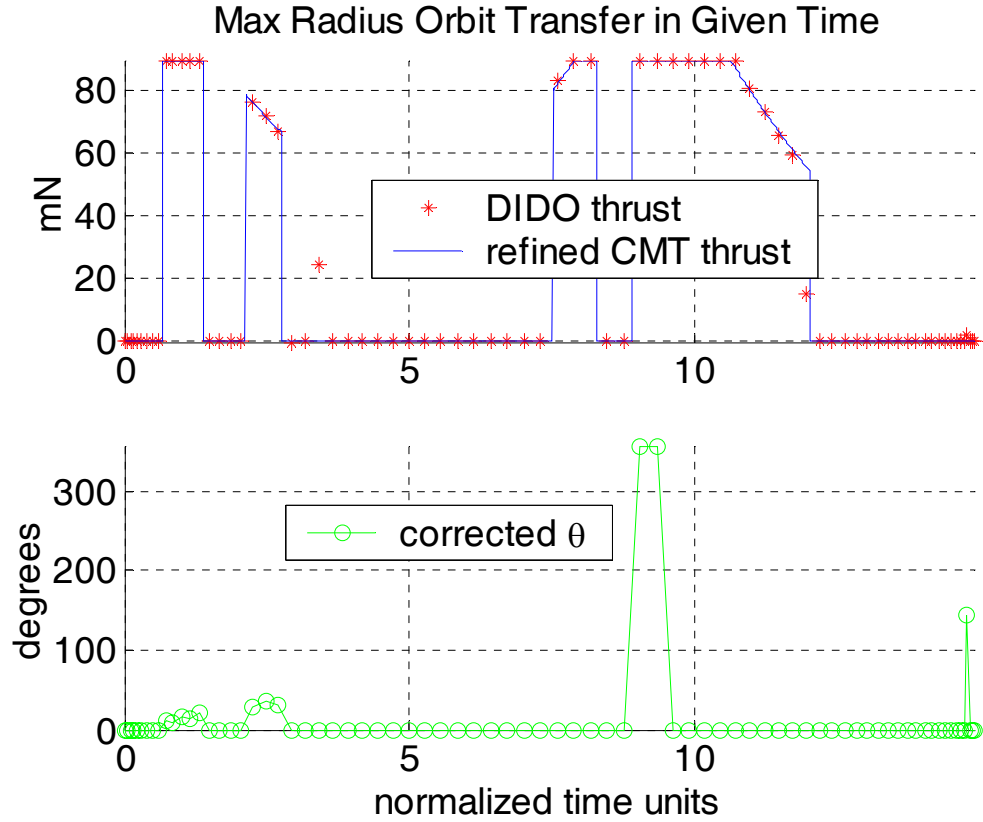

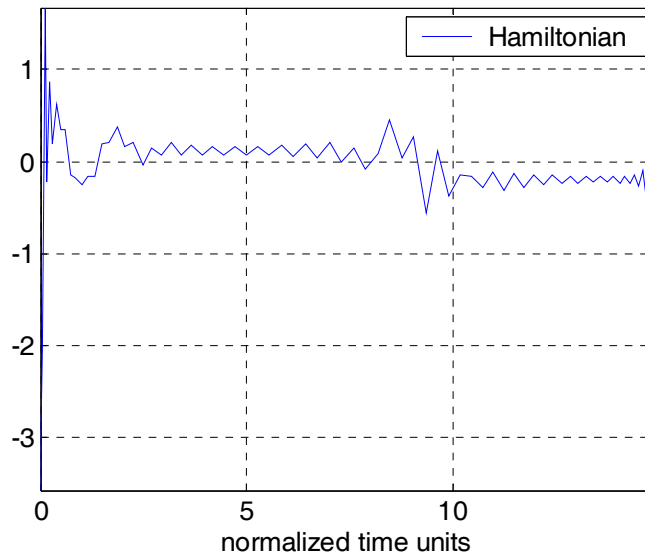Figure 34   Elliptical Rendezvous Control History (Result 2)

Figure 35   Elliptical Rendezvous Hamiltonian (Result 2)

There are four extremal cases of this Earth to Asteroid trajectories: max a, min a, max e, and min e. Since the min e case ($e=0$) is simply the circular orbit, none of which are occupied by asteroids this will not be examined here. The remaining cases, max a, min a, and max e, will be shown here for examples of differing trajectories. The only difference in the code between all these elliptical cases is the cost function. For the orbit and optimal control history shown in Figure 36 and Figure 37, the following cost function was used:

$$J = \alpha a_f + \beta e_f \quad \text{where: } \alpha = -1,\ \beta = -0.0001 \tag{78}$$

This will maximize the final semi-major axis. The weight β could be set to zero, however for an unexplained reason, DIDO runs faster with it none zero.
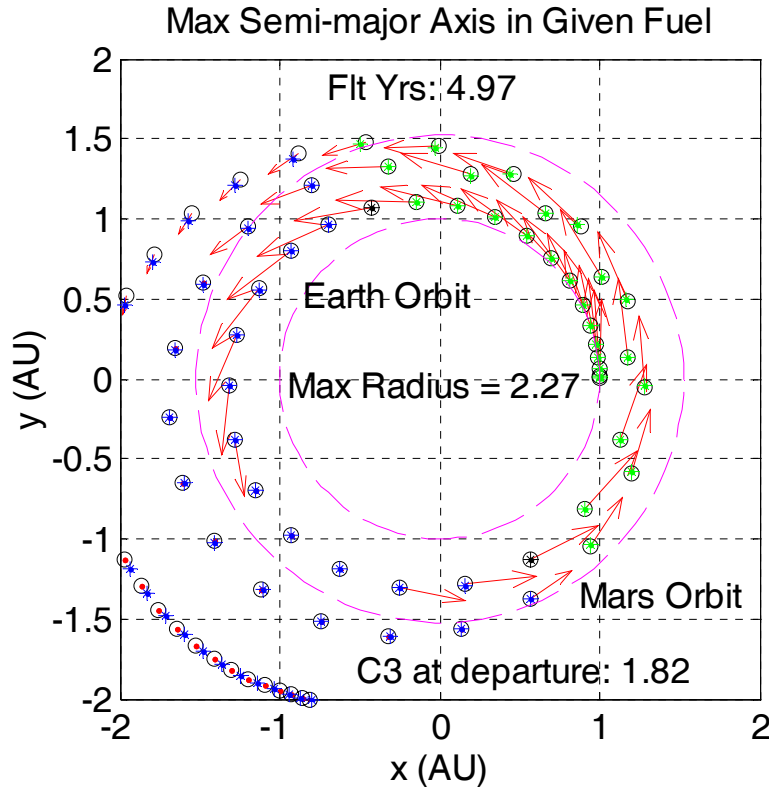
Figure 36   Maximum a Elliptical Rendezvous (Result 3)

This result appears to be both feasible and optimal. It has a flat Hamiltonian, shown in Figure 38, with a mean value of 0.055 and controls that satisfies the KKT requirements. It propagates best with ODE23t with a final $a = 1.82$, $e = 0.253$, and maximum radius from the sun of 2.27 AU.

64

Figure 37   Maximum *a* Elliptical Rendezvous Controls (Result 3)
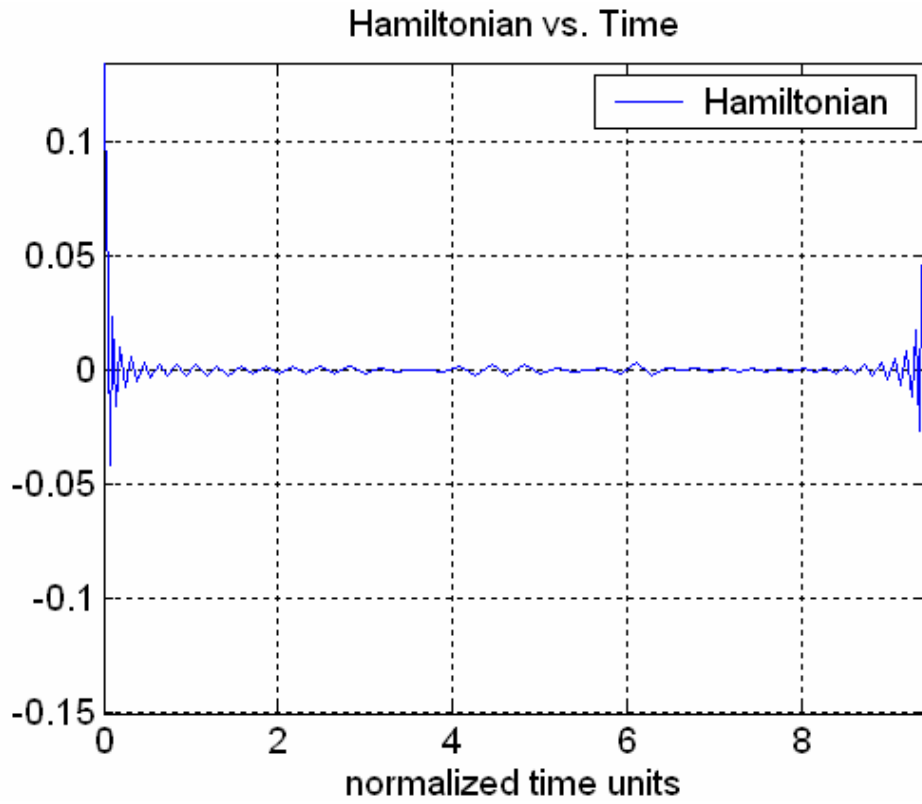
Figure 38   Maximum *a* Elliptical Rendezvous Hamiltonian (Result 3)

In the maximum eccentricity case, the orbit and optimal control history are shown in Figure 39 and Figure 40, and the following cost function was used:

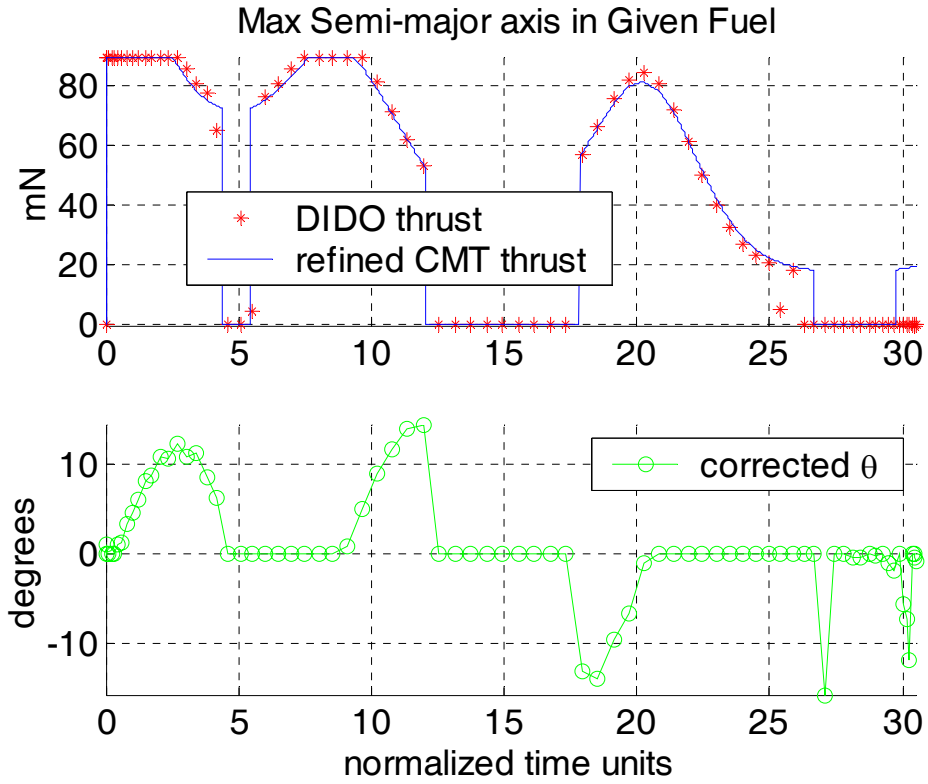$$J = \alpha a_f + \beta e_f \quad \text{where: } \alpha = -0.0001, \ \beta = -1.0 \tag{79}$$

Again, the unneeded weight could be set to zero. Similarly, this result appears to be both feasible and optimal. It has a flat Hamiltonian, as shown in Figure 41 with a mean value of -0.0222 and controls that satisfies the KKT requirements. It propagates best with ODE113 with a final $a = 1.5$, $e = 0.384$, and maximum radius from the sun of 2.07 AU. One interesting difference between the two cases is that to maximize eccentricity, a lot of off-axial thrusting is done.

Figure 39   Maximum *e* Elliptical Rendezvous Orbit (Result 4)
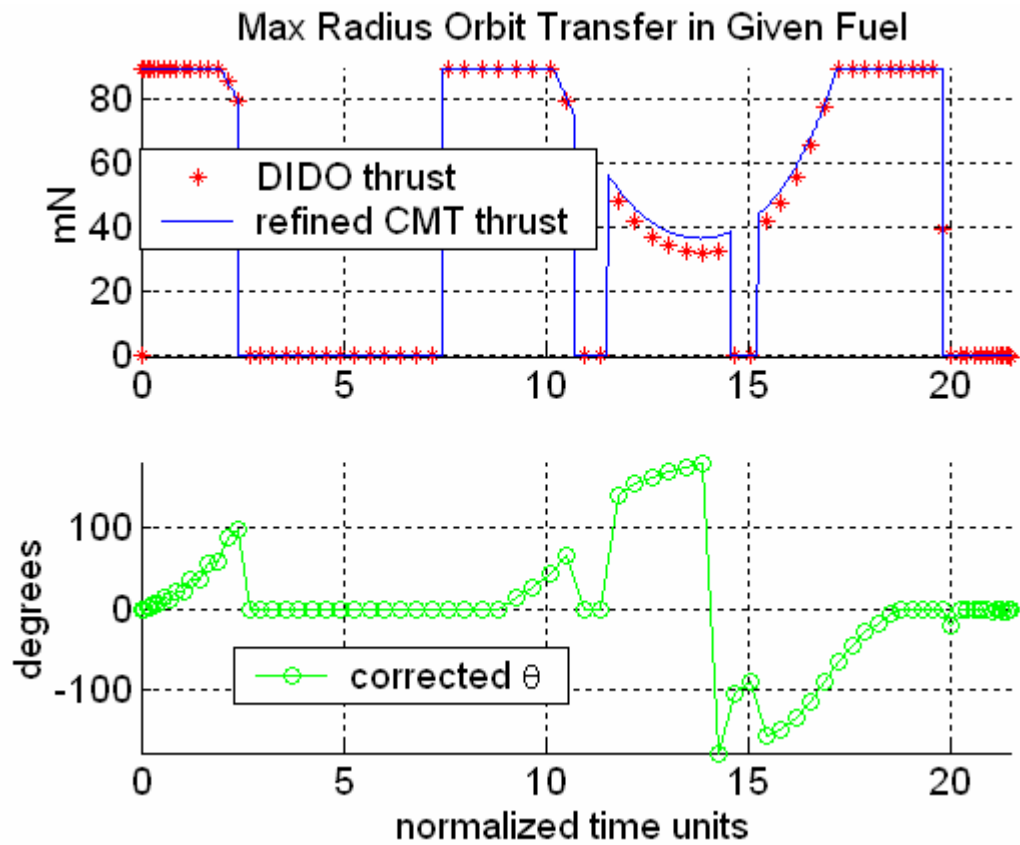
Figure 40   Maximum *e* Elliptical Rendezvous Controls (Result 4)
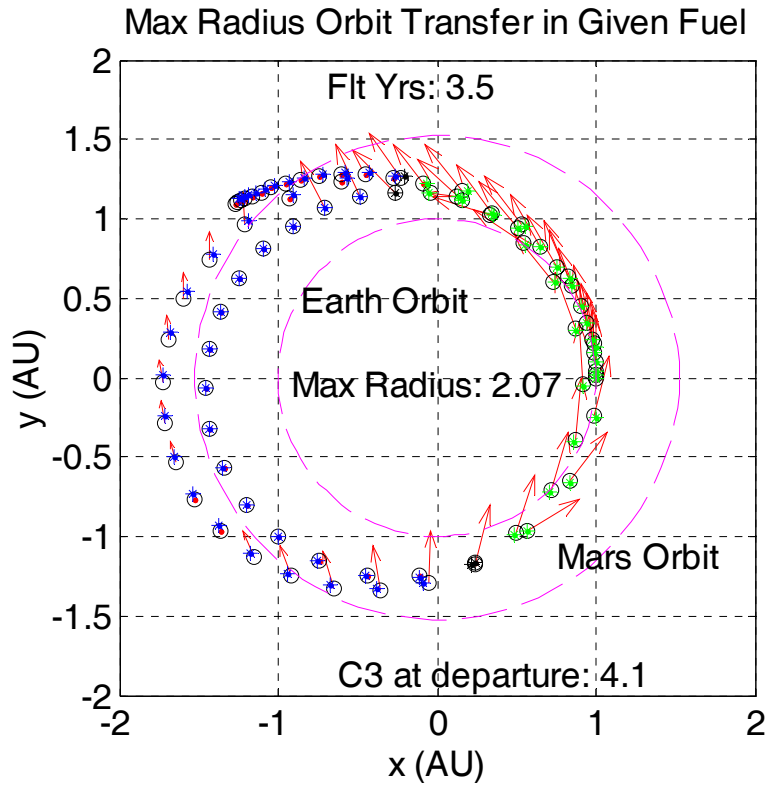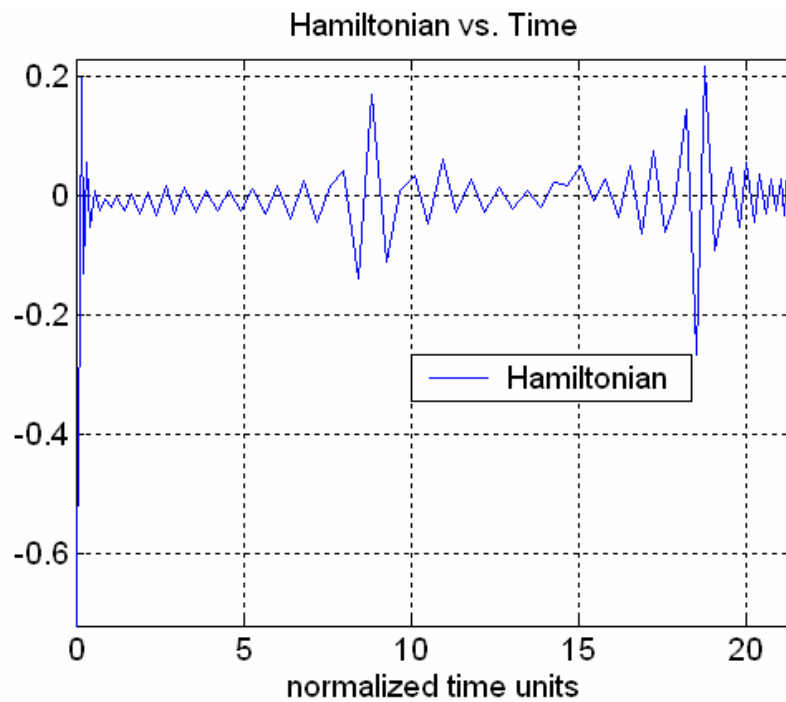

Figure 41   Maximum *e* Elliptical Rendezvous Hamiltonian (Result 4)

67

The final case is the minimum semi-major axis (*a*) case, the orbit and optimal control history are shown in Figure 42 and Figure 43 and the following cost function was used:

$$J = \alpha a_f + \beta e_f \quad \text{where: } \alpha = -1, \ \beta = -0.00001 \quad (80)$$

Similarly, this result appears to be both feasible and optimal. It had a flat Hamiltonian, Figure 44, with a mean value of -0.0079 and controls that satisfies the KKT requirements. It propagates best with ODE15s with a final *a* = 0.673, *e* = 0.0677, and minimum radius from the sun of 0.599 AU. As expected, most of the thrusting is in the opposite direction of motion to slow the spacecraft. This trajectory has slightly higher error between the DIDO solution and propagated trajectory, but it is still reasonable. Also, this DIDO sun took over 44 minutes longer than all the previous.

Figure 42   Minimum *a* Elliptical Rendezvous Orbit (Result 5)

Figure 43   Minimum *a* Elliptical Rendezvous Controls (Result 5)

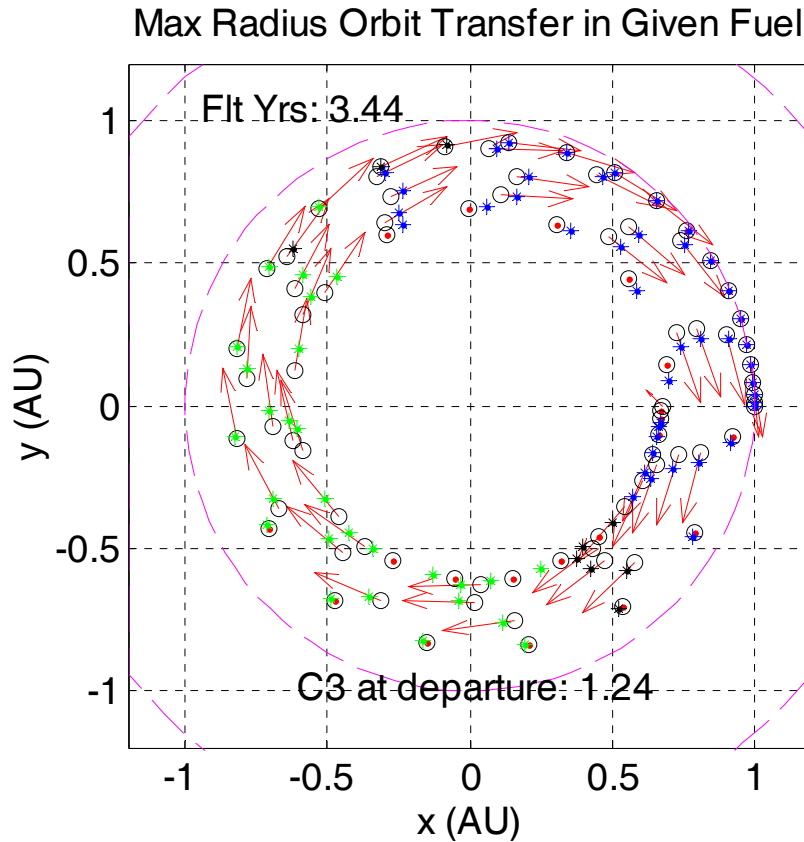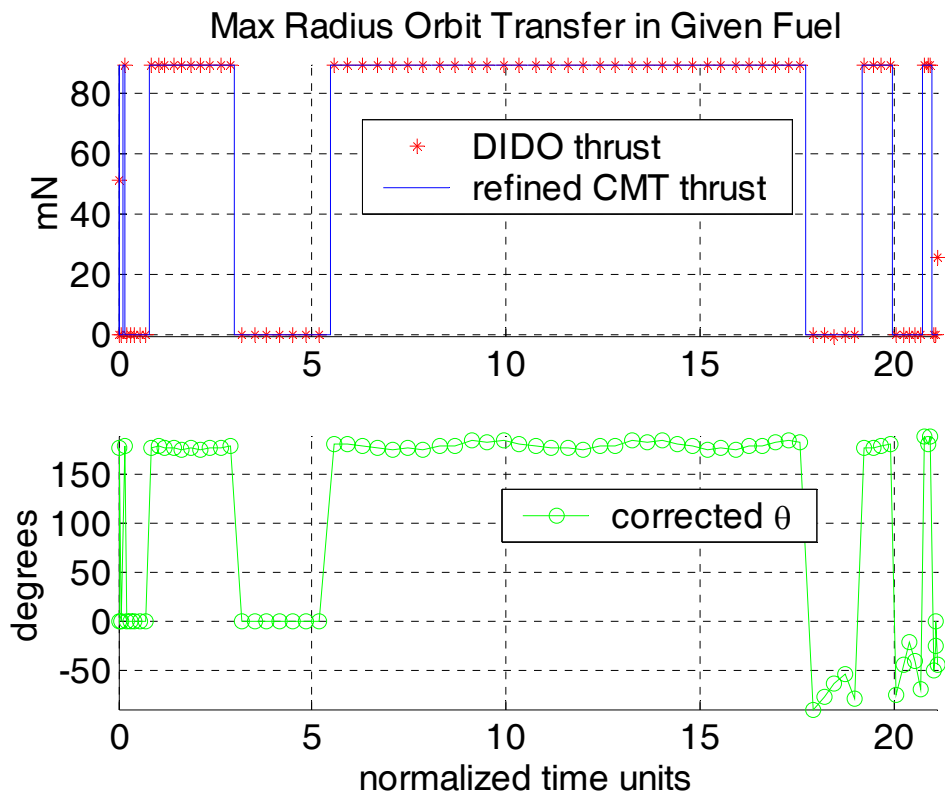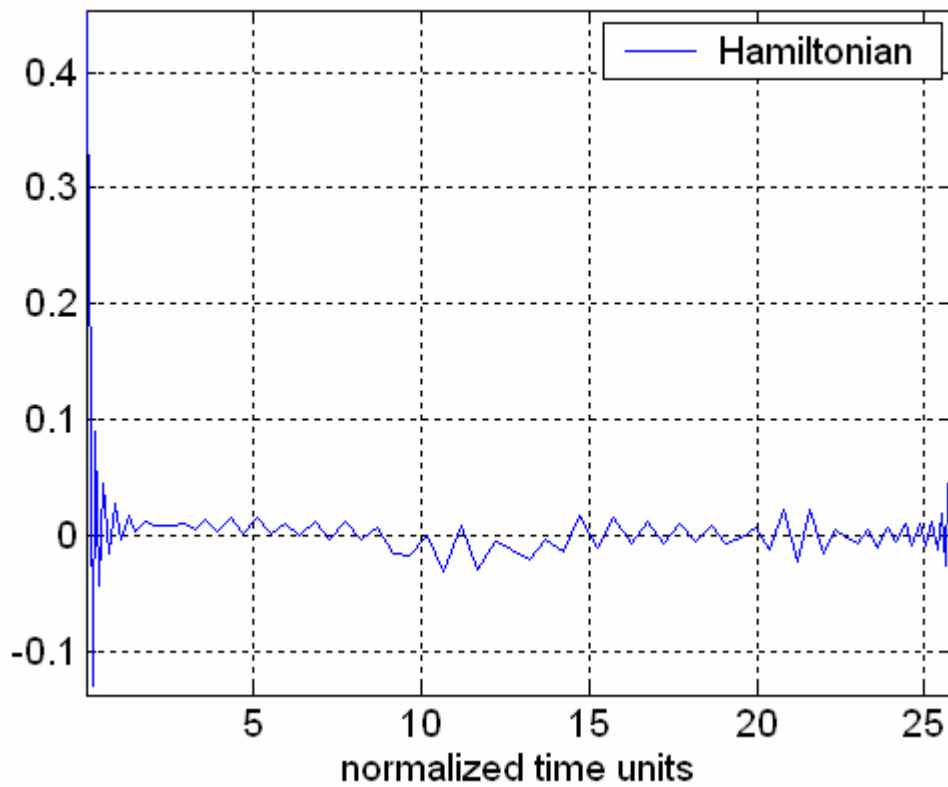Figure 44   Minimum *a* Elliptical Rendezvous Hamiltonian (Result 5)

69

Objective	Semi-major Axis, a (AU)	Eccentricity, e
Maximize a (Result 3)	1.82	0.253
Minimize a (Result 5)	0.673	0.0677
Maximize e (Result 4)	1.5	0.384
Minimize e (circular case)	1.65 – 0.884 [7]	0
Mix of weights (not shown)	0.8161	0.272

Table 4.   Summary of Results for Asteroid Rendezvous

E.   EARTH TO ASTEROID REACHABLE SET

Using the data from the optimization routines just presented several permutations of the reachable set will be developed. First, an "inner approximation" will be easily constructed using the four extremal objectives. Since, the trajectory starts at a minimum eccentricity (Earth circular orbit) only three DIDO runs where needed to construct Figure 45. This two-dimensional projection of asteroid orbit energies into the a-e plane combined with the convex polytope of the extremal solutions represents the reachable set that has been mathematically proven to be reachable (for a given level of fidelity). The inner approximation can be automatically constructed for any optimization routine since the weights on the cost function are predetermined to be the set of $(\alpha, \beta) \in \{(1,0), (-1,0), (0,1), (0,-1)\}$.

---

[7] This result is not shown but included in table for completeness. Only requires changing sign of circular case cost function.
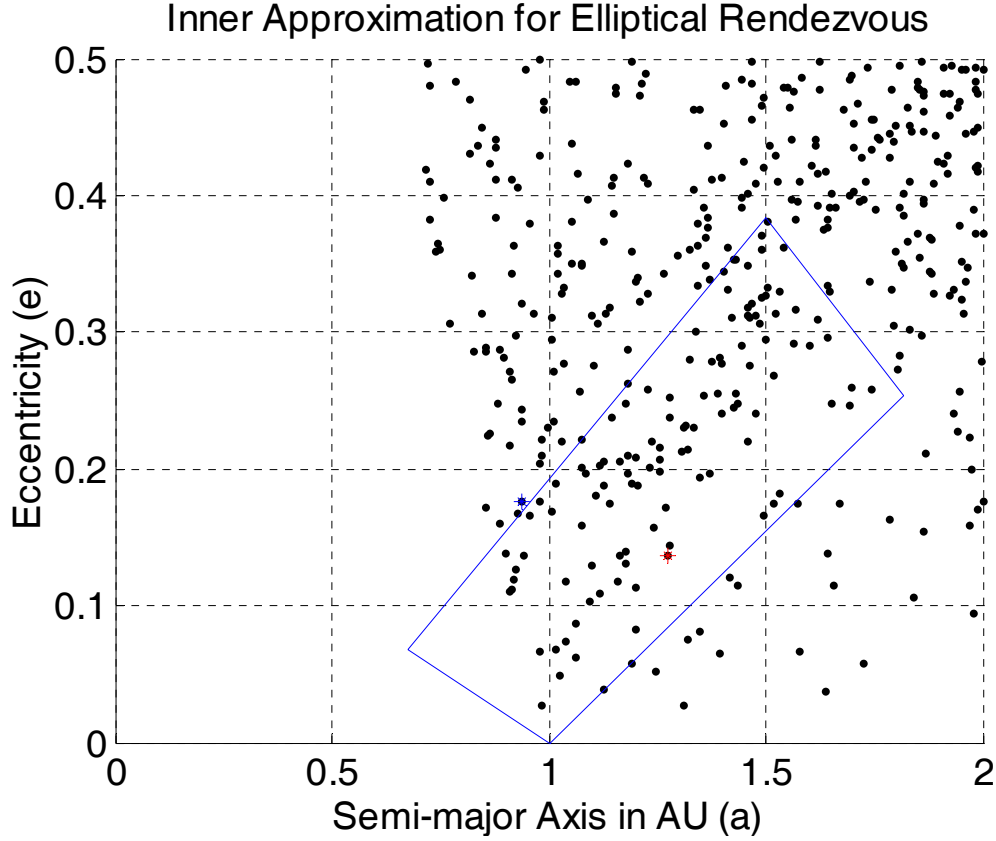
Figure 45   Inner Approximation of Elliptical Rendezvous

The more complete outline of the reachable set can be constructed using all the points listed in Table 4. This 6-point representation of Figure 46 on the *a-e* domain encompasses much more area. The three new points on this figure are all optimal solutions for a combination of maximizing, or minimizing, the final elliptical orbit of the spacecraft. Since the minimizing *e* case was trivial, i.e. started at Earth with a minimum *e*, one could wonder how *a* would vary for the minimum *e* case (what the bottom of an inner approximation looks like). This was done holding *e* constant at zero and rerunning the min and max *a* optimization routines ($\alpha = [-1,1]$ *and* e_f constrained to zero or circular case). Similarly, the "Mix *a/e*" case was just an educated guess at the weights which would provide a boundary point of the reachable set in that region ($\alpha = .5$, $\beta = -1$). While seemingly providing more detail in the solution, this process can not as easily be automated since it required at least one judgment on the weight.
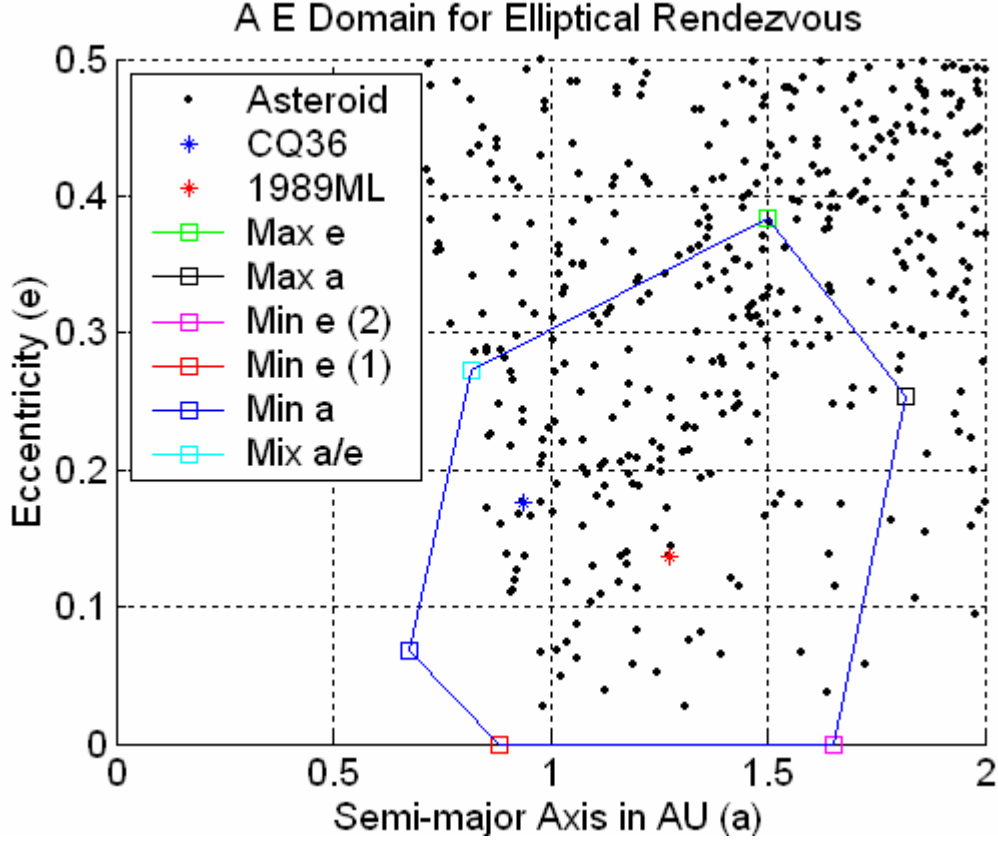
Figure 46   Multi-point Reachable Set

Adding boundary points by varying the weights between -1 and 1 can be done ad infinitum. In theory, this would provide a continuous boundary for the actual reachable set, however would also take an infinite amount of time. When adding a third dimension, such as for inclination, the process to determine an inner approximation would only take one more optimization run (max i), since the Earth is at the minimum inclination case. To construct more detailed boundaries, such as in Figure 46, a third dimension would need to include many more optimization runs, depending on the fidelity of the boundary required.

To include more of the possible targets in the reachable set than the inner approximation, using only the extremal cases, an "outer approximation" can be constructed, as in Figure 47. An outer approximation is the boundary where asteroids are within the set $\{a_{target} \leq a_{max} \cap a_{target} \geq a_{min} \cap e_{target} \leq e_{max} \cap e_{target} \geq e_{min}\}$.   The targets

outside this boundary are mathematically required to be unreachable. Thus, with only the extremal solutions to trajectory requirements, the target set is greatly reduced.
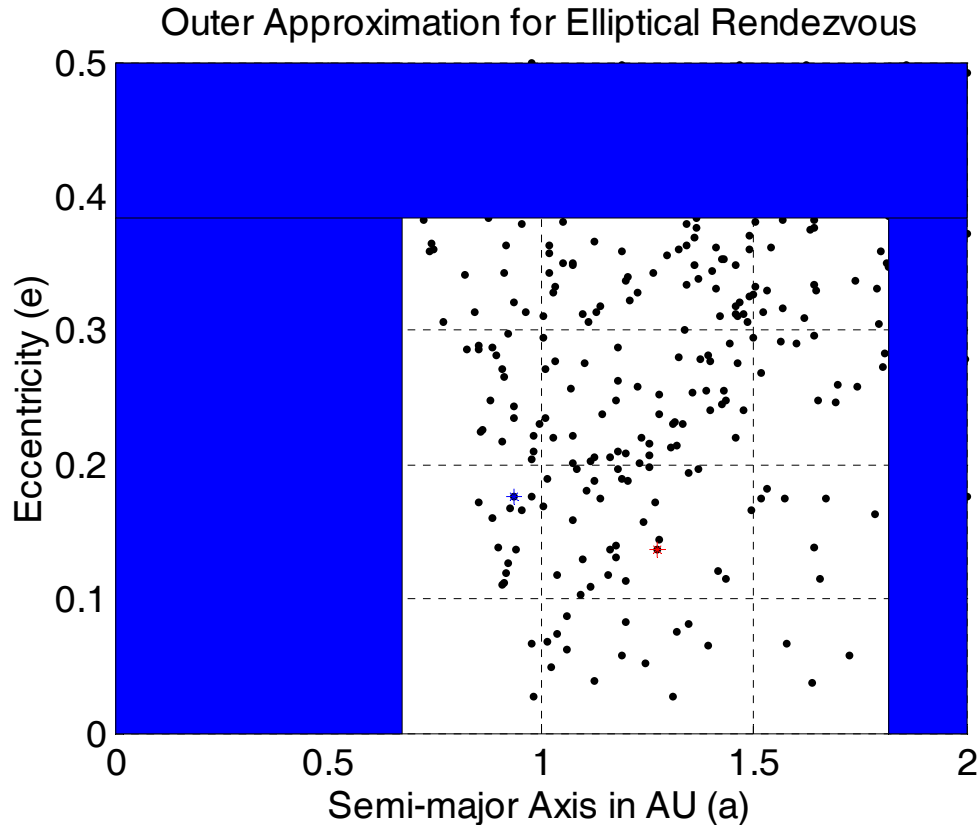
Figure 47   Outer Approximation of Elliptical Rendezvous

Figure 48 shows all the previous representations on one plot. This includes a region of known reachablity, known unreachability, and the possible benefits of providing more detail to give a better idea of the shape of the actual reachable set. Using the concept of inner and outer approximations, the effort to screen targets as possibilities becomes much easier and then the remaining candidates can be specifically targeted to examine mission feasibility.   The inner and outer approximations can be extended to beyond two dimensions without adding many additional optimizations runs, just problem fidelity, to further reduce the possible candidates. This methodology can be useful to provide mission planners a reduced set of possible targets to be screened for reasons other than trajectory feasibility, such as risk reduction or scientific suitability. For the 6157 targets plotted in Figure 17, these results for an asteroid rendezvous mission would limit

the possible target set down to 720, or almost an order of magnitude. If inclination is limited to < 10°, the number of possible targets is further reduced to 132.

Figure 48   Inner and Outer approximations

The circular and elliptical rendezvous case were simple examples of how to formulate a NLP for optimization, show feasibility, show local optimality, and how to use reachable sets to limit the number of possible targets for further analysis. The following chapters will add detail to more interesting cases for sample return missions.
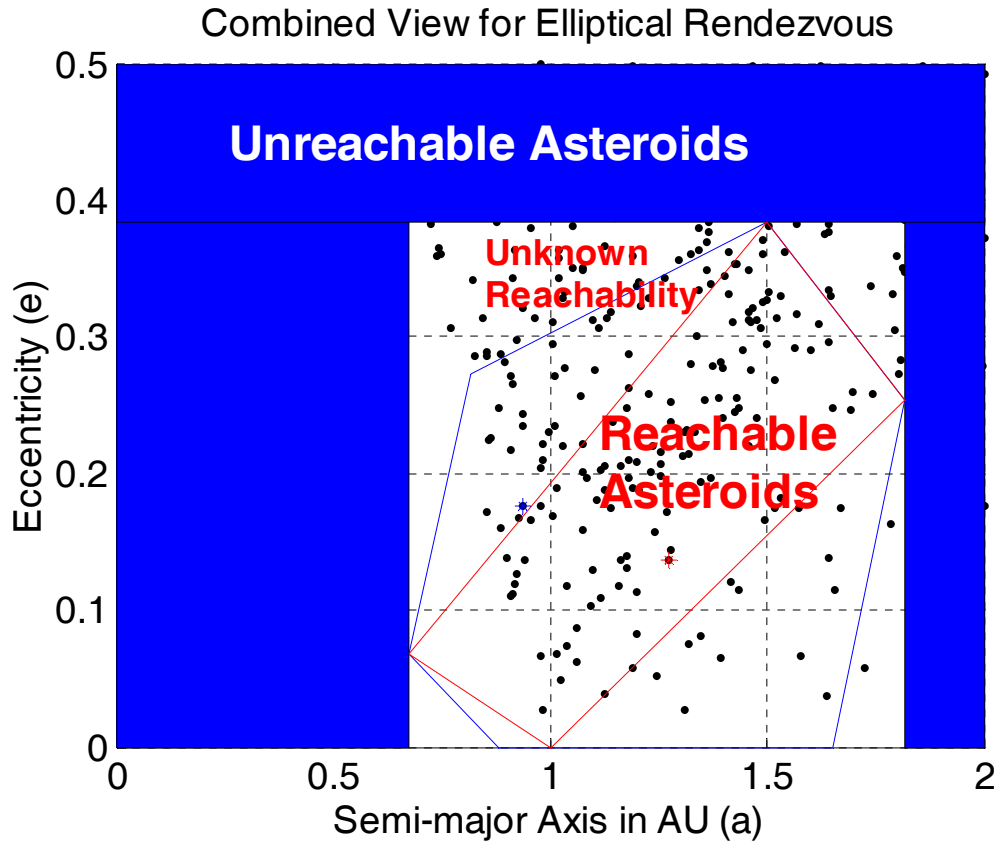
V.   RENDEZVOUS AND RETURN

A.   DEPART ASTEROID, INTERCEPT EARTH

This problem could logically be next step to examine a combined problem of optimizing both the trips to and from the asteroid independently and then joining the result together afterward. However, since the fuel needs to be shared between the two trajectories and the stay time should be optimized, this chapter will perform one optimization routine for the trip from Earth to the asteroid and then return.

B.   DEPART EARTH, RENDEZVOUS WITH ASTEROID, RETURN TO EARTH

Figure 49 shows a basic representation of this problem. The spacecraft will depart Earth at time t_0, make the necessary orbit maneuvers to rendezvous at time t_i, stay for a period not less than 90 days, depart the asteroid at time t_{i+1}, and then return for and Earth flyby to drop off the sample at time t_f. As before, the C3 departure energy from Earth will be optimized and the spacecraft will expend all the fuel to maximize the asteroid orbit parameters, *a* and *e*. During the stay time at the asteroid, where the spacecraft will map out the surface and gravitational field in preparation for the actual sample collection process, it is assumed the spacecraft will expend 20 kg of fuel. The return will be an Earth flyby with the V_∞ of arrival constrained to less than 5 km/s to minimize impact of the sample drop-off. Again, this case will be studied in two dimensions with the same propulsion model and limitations used in the Elliptical Rendezvous. Neither atmospheric effects, nor return declination requirements to hit the Utah Test and Training Range were studied in the Earth return profiles.

For every solution, the entire scenario is optimized together, however the first and second legs of the trajectories are shown separately for readability. As before, the extremal cases of the cost function will be explored to develop inner and outer approximations.
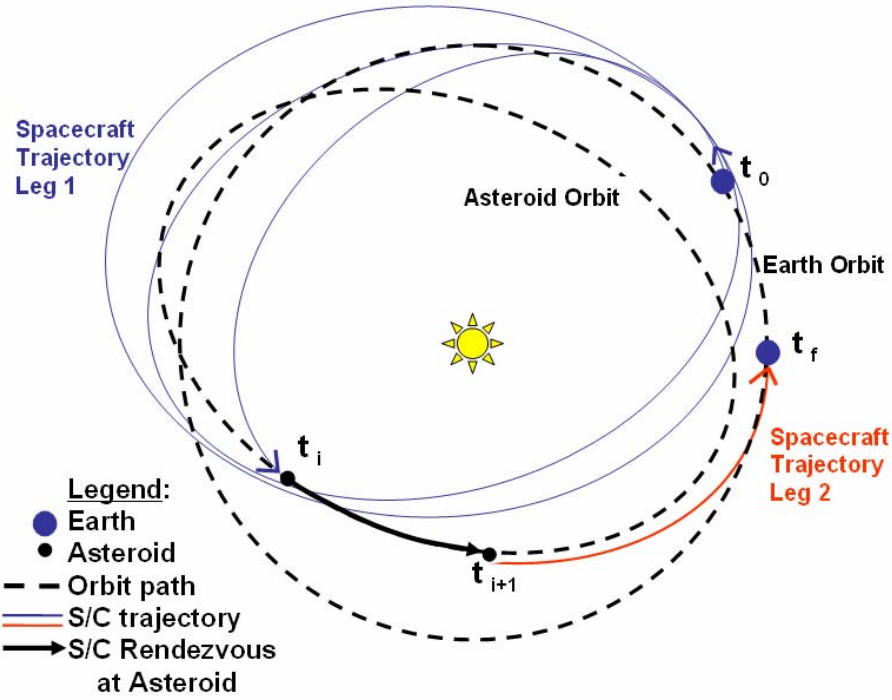
Figure 49   Asteroid Rendezvous and Return Low Thrust Trajectory Representation

1.   Rendezvous and Return Problem Formulation

This optimization run computes trajectories for Earth to asteroid rendezvous' and return missions. The dynamics remains the same as use previously however the since the objective to examine the asteroid orbit extremal cases, the cost function is slightly modified to account for the orbit at the time of rendezvous, or:

$$J = \alpha a_i + \beta e_i \qquad \text{where: } \alpha + \beta \text{ are weights} \qquad (81)$$

The time t_i, is the time at node i, or the number of the node (out of 80 used) that the rendezvous first occurs. The next node, at time t_{i+1}, is the point of departure from the rendezvous after the optimized stay time. Thus, the cost function could also be described as follows since the spacecraft is in the same orbit during the rendezvous period.

$$J = \alpha a_{i+1} + \beta e_{i+1} \qquad (82)$$

In order to accomplish this optimization within DIDO, a "knot" [Ref. 7] must be used at the node i. This allows the program to designate conditions that must be met at that point, whether it is the 2nd or the 79th node of the solution. Thus, these constraints are

76

events since they happen at a specific point, vice path constraints that are met throughout the trajectory. In the previous problems, events only occurred at boundary points, i.e. start or end points.

The event constraints are set to a range of inequalities (with the lower bound to the left and upper bound to the right in the set). There is more than twice the number of event constraints; many of the new events are to ensure that state continuity on either side of the knot. As before, any non-constrained states or conditions are considered free and the initial mass state is set dependant on a function of the C_3 boost optimized for launch. Don't confuse the angular transfer state, nu or ν, in the 8th constraint with the radial and transverse states of v_r and v_t. The 8th constraint ensures that rendezvous arrival and departure angular transfer state changes by the same number of radians as the mean motion of the asteroid. The 9th constraint ensures a mass loss for the near asteroid maneuvers. The 12th constraint ensures the flyby occurs at Earth and accounts for the mean motion of Earth. The 13th constraint is the limit to ensure the flyby is less than 5 km/s.

$$
\mathbf{e}_{[l,u]} =
\begin{bmatrix}
r_0 \\
\theta_0 \\
m_0 \\
\left(v_{t_0} - 1\right)^2 + v_{r_0}^2 - C_3 \\
r_i - r_{i+1} \\
v_{r_i} - v_{r_{i+1}} \\
v_{t_i} - v_{t_{i+1}} \\
\left(\nu_{i+1} - \nu_i\right)/n_a \\
m_i - m_{i+1} - 20\ kg \\
e_i \\
r_f \\
\nu_f - (t_f - t_0)/n_e \\
\left(v_{t_f} - \sqrt{\mu_e/r_f}\right)^2 + v_{r_f}^2 \\
m_f
\end{bmatrix}
=
\begin{bmatrix}
[1,1] \\
[0,0] \\
f(C_3) \\
[0,0] \\
[0,0] \\
[0,0] \\
[0,0] \\
[90,185\ days] \\
[0,0] \\
[0,1] \\
[1,1] \\
[0,0] \\
[0,25\ km^2/s^2] \\
[M_{dry}, M_{dry}]
\end{bmatrix}
\tag{83}
$$

The path constraint and engine model are the same, thus the Hamiltonian and for the Thrust switching function, remains unchanged. Unfortunately, the DIDO version 2003b1c used in this thesis does not return dual variables in problems containing knots. Thus the costates, Hamiltonian, and all LaGrange multipliers are unavailable to check the CMT and KKT conditions to confirm optimality as before.

a. Maximize Semi-majorAxis ($\alpha = -1, \ \beta = 0$)

Figure 50 through Figure 54 shows the state, control and orbit history of a optimal trajectory for the extremal case of maximizing a. The control history was propagated with ODE45 and shows very small deviations from DIDO states and thus proves feasible since no constraints were broken. The final result was the asteroid orbit it rendezvoused with for a period of 103 days had an a of 1.2887 and e of 0.1150. This solution was deemed "probably optimal" by DIDO and only took 5.2 minutes to solve.
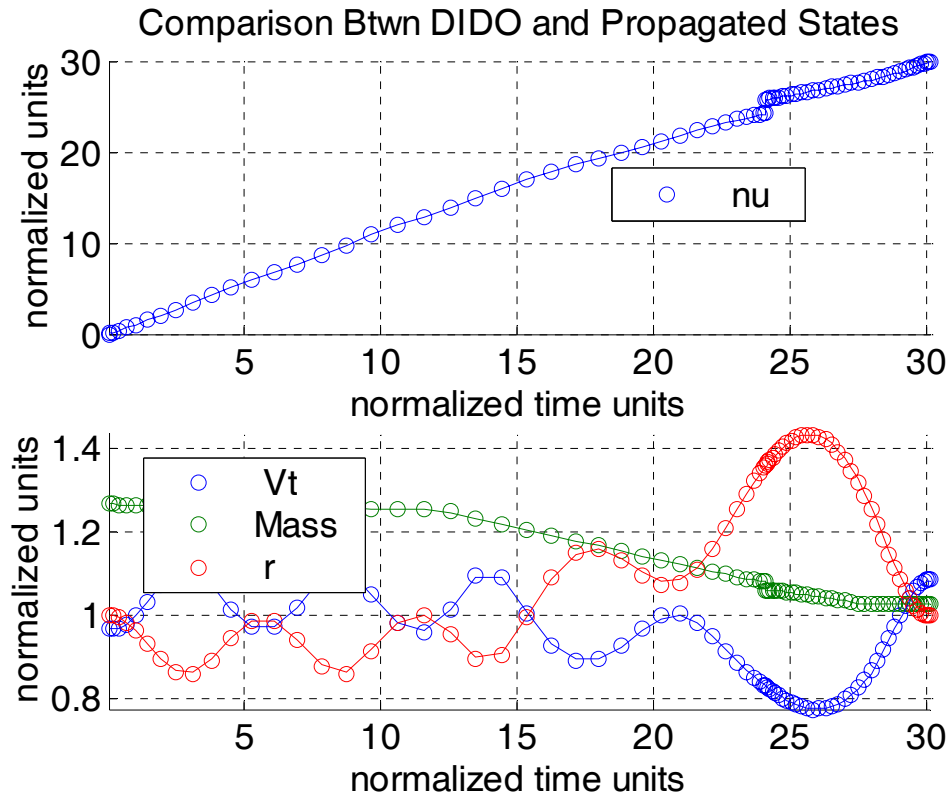

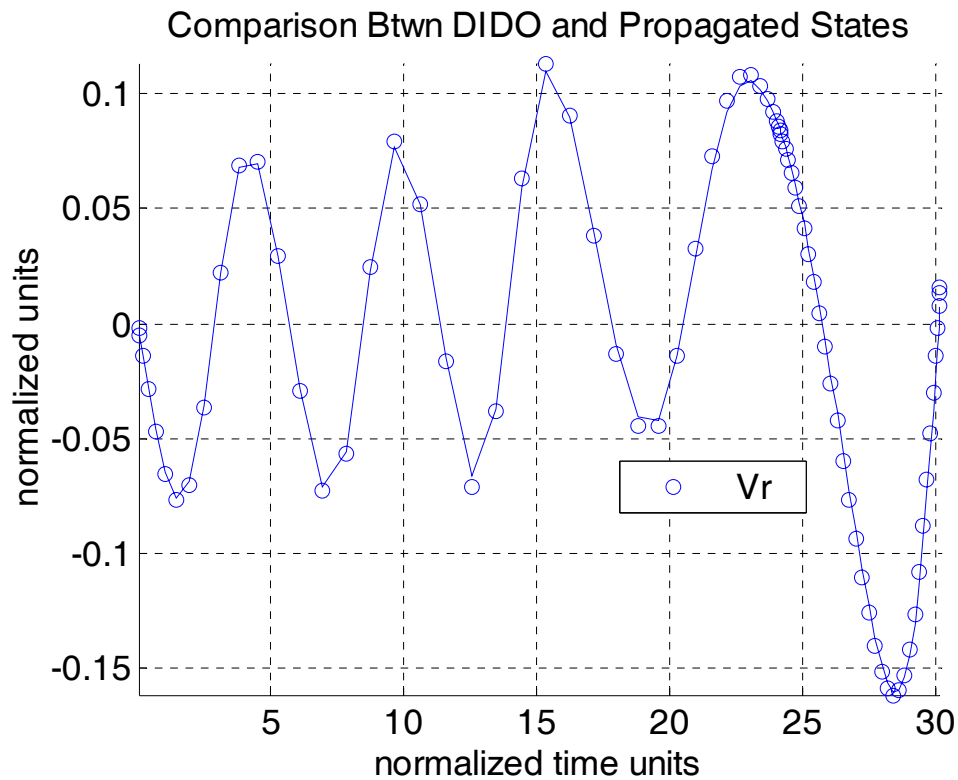Figure 50   Rendezvous and Return Maximize *a* States (Result 6)

Figure 51   Rendezvous and Return Maximize *a* States continued (Result 6)
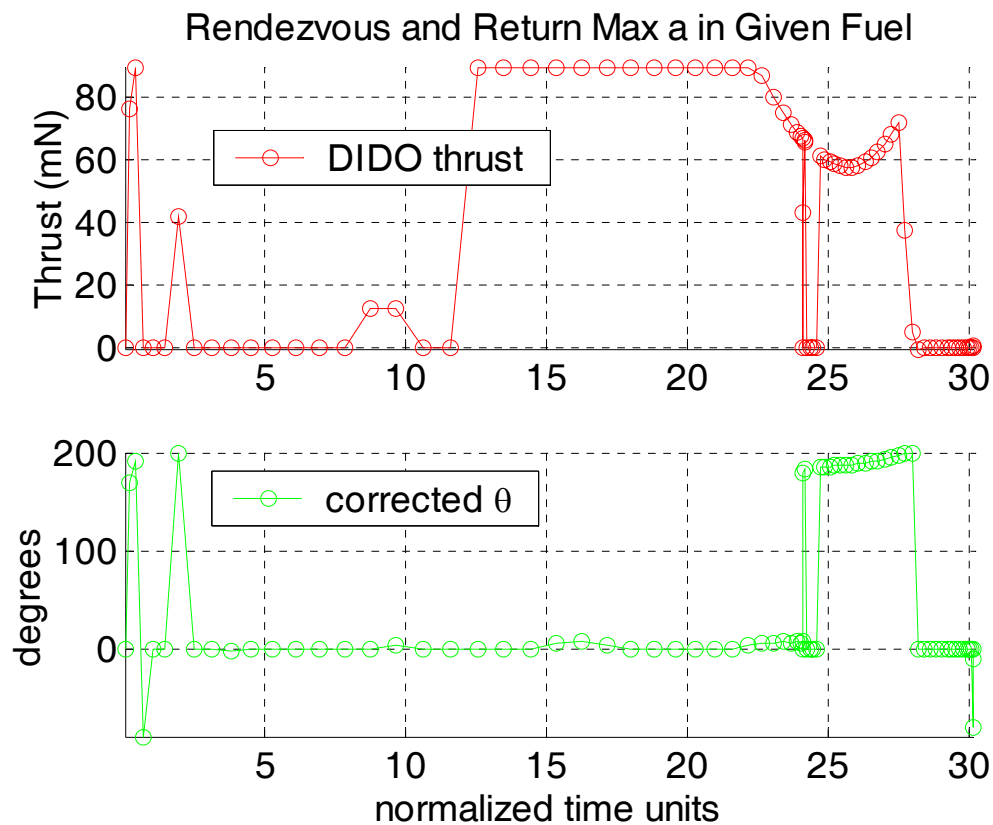

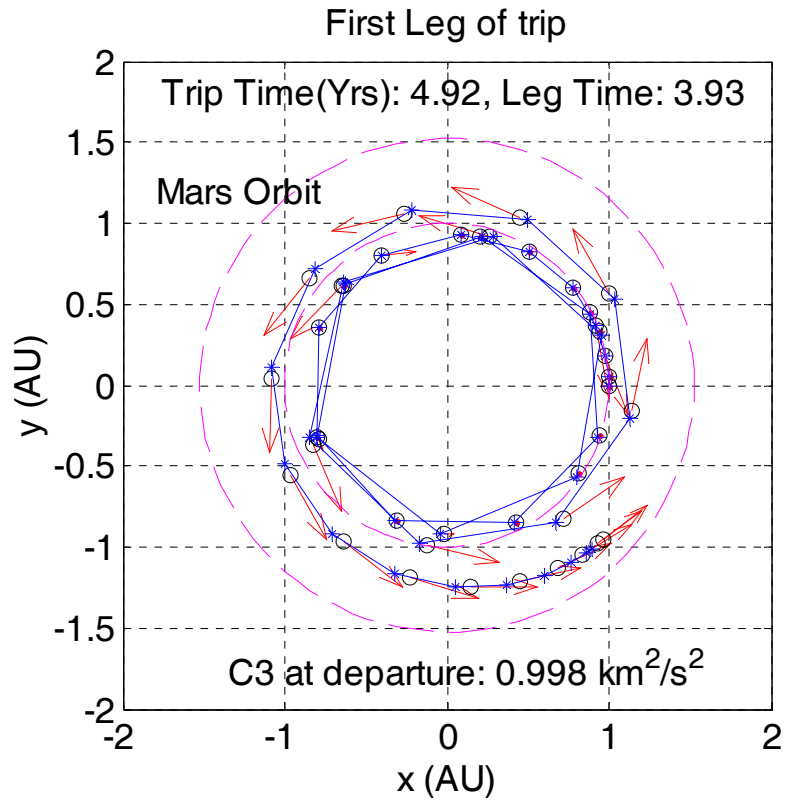Figure 52   Rendezvous and Return Maximize *a* Control History (Result 6)

79

Figure 53   Rendezvous and Return Maximize *a,* Earth to Asteroid Plot (Result 6)
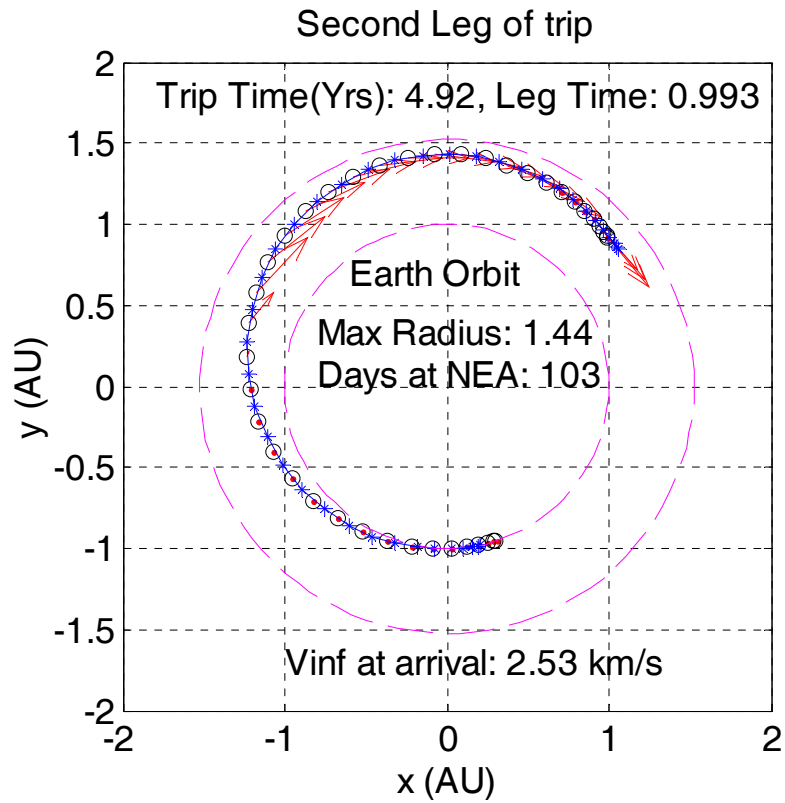
Figure 54   Rendezvous and Return Maximize *a,* Asteroid to Earht Plot (Result 6)

The start and end points shown in the previous orbit depictions are easier to read if one remembers that the motion is always in a prograde direction (counter-clockwise). Also, as previous mentioned, the LGL points used in DIDO bunches near the beginning, end, and near knots, such as the asteroid rendezvous point to increase accuracy of the solution. Since the degree of correlation between the DIDO solution and the propagated control history to verify feasibility can almost as easily be seen on the orbit plots as well as the state history plots, the state history plots will not be shown for simplification purposes.

b. *Minimize Semi-major Axis ($\alpha = 1$, $\beta = 0$)*

Figure 55, Figure 56, and Figure 57 shows the control and orbit history of an optimal trajectory for the extremal case of minimizing *a*. The control history was propagated with ODE23s and also has a very small error from the DIDO solution and thus proves feasible since no constraints were broken. The final result was the asteroid orbit it rendezvoused with for a period of 98.7 days had an *a* of 0.8607 and *e* of 0.1686. This solution was deemed "probably optimal" by DIDO and only took 6.8 minutes to solve.
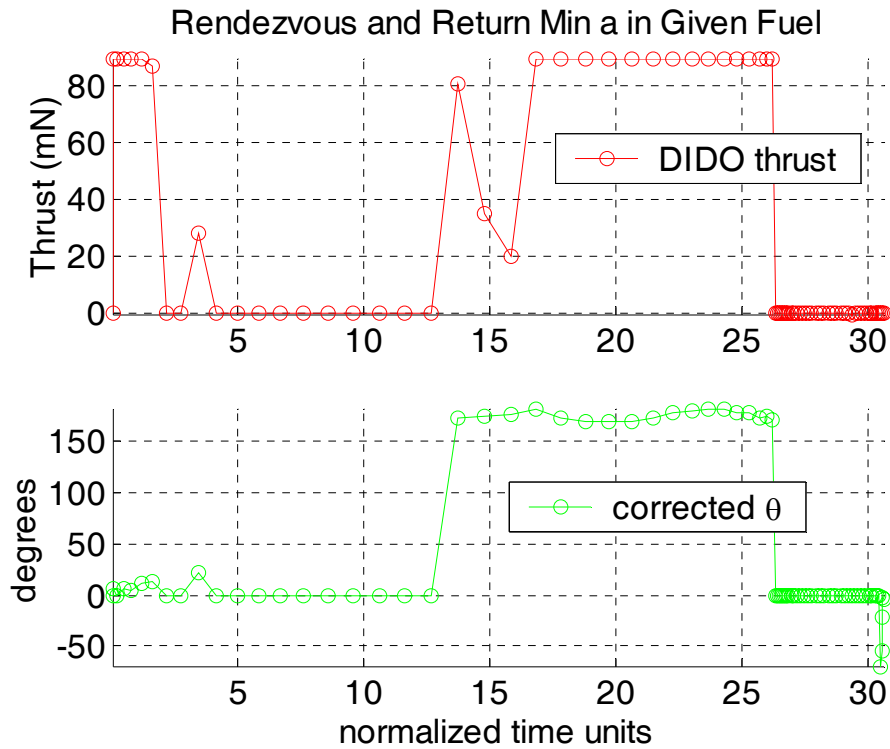
Figure 55   Rendezvous and Return Minimize *a*, Control History (Result 7)

First Leg of trip

Trip Time(Yrs): 4.99, Leg Time: 4.31

Mars Orbit

C3 at departure: 1.76 km²/s²
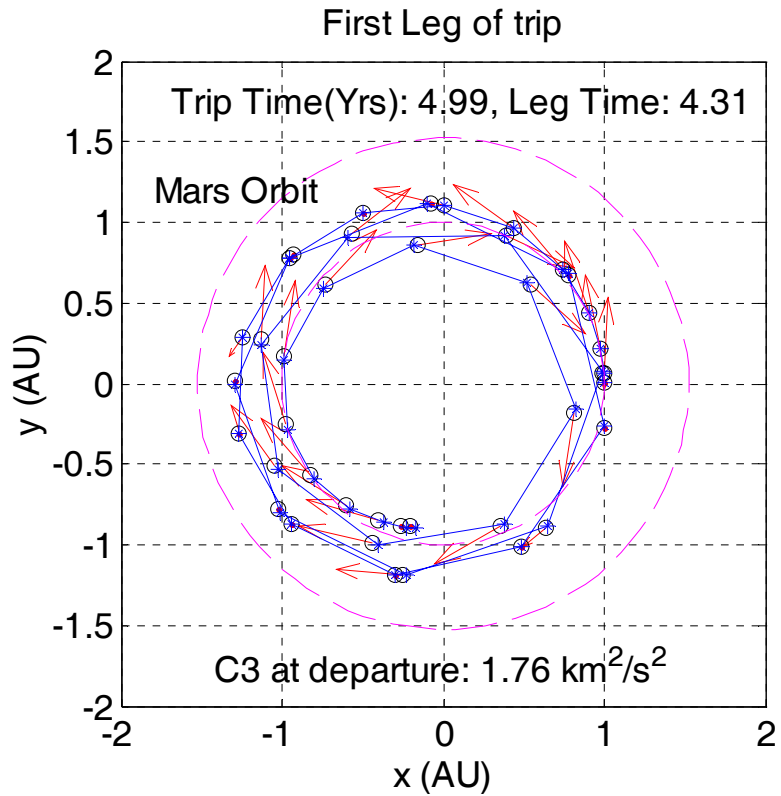
Figure 56   Rendezvous and Return Minimize *a,* Earth to Asteroid Plot (Result 7)

Second Leg of trip

Trip Time(Yrs): 4.99, Leg Time: 0.683

Max Radius: 1.01
Days at NEA: 98.7

Vinf at arrival: 2.86 km/s
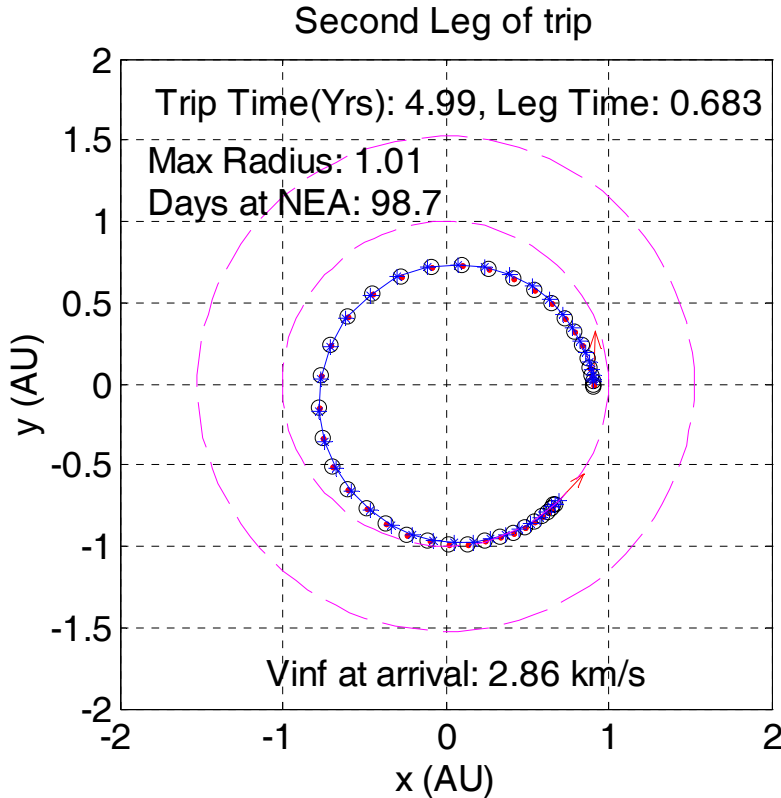
Figure 57   Rendezvous and Return Minimize *a,* Asteroid to Earth Plot (Result 7)

c.   ***Maximize Eccentricity*** *($\alpha = 0.00001$, $\beta = -1$)*

Figure 58, Figure 59, and Figure 60 shows the control and orbit history of an optimal trajectory for the extremal case of maximizing *e*. The control history was propagated with ODE15s and also has a very small error from the DIDO solution and thus proves feasible since no constraints were broken. The final result was the asteroid orbit it rendezvoused with for a period of 185 days had an *a* of 1.099 and *e* of 0.267. This solution was deemed "probably optimal" by DIDO and only took 24.6 minutes to solve. This was most difficult to solve and was sensitive to small changes in guesses, stay time ranges, and cost function weights. Consequently, to obtain a feasible and optimal solution for this case, a small non-zero value for the weight on *a* was required.
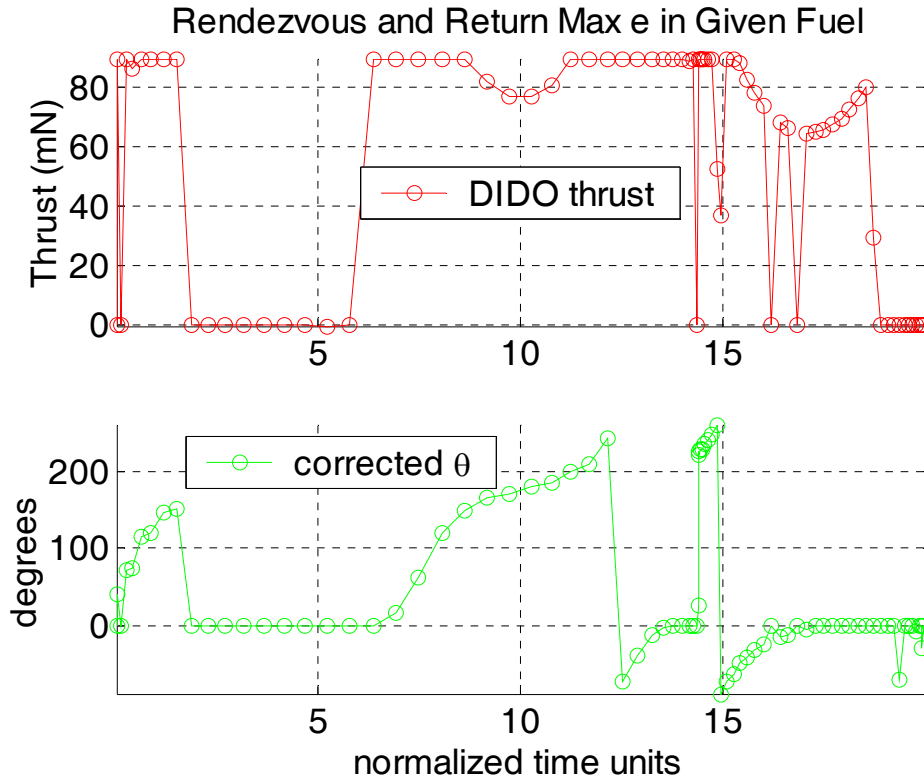
Figure 58   Rendezvous and Return Maximize *e,* Control History (Result 8)

First Leg of trip

Trip Time(Yrs): 3.76, Leg Time: 2.35

Mars Orbit

C3 at departure: 2.2 km^2/s^2
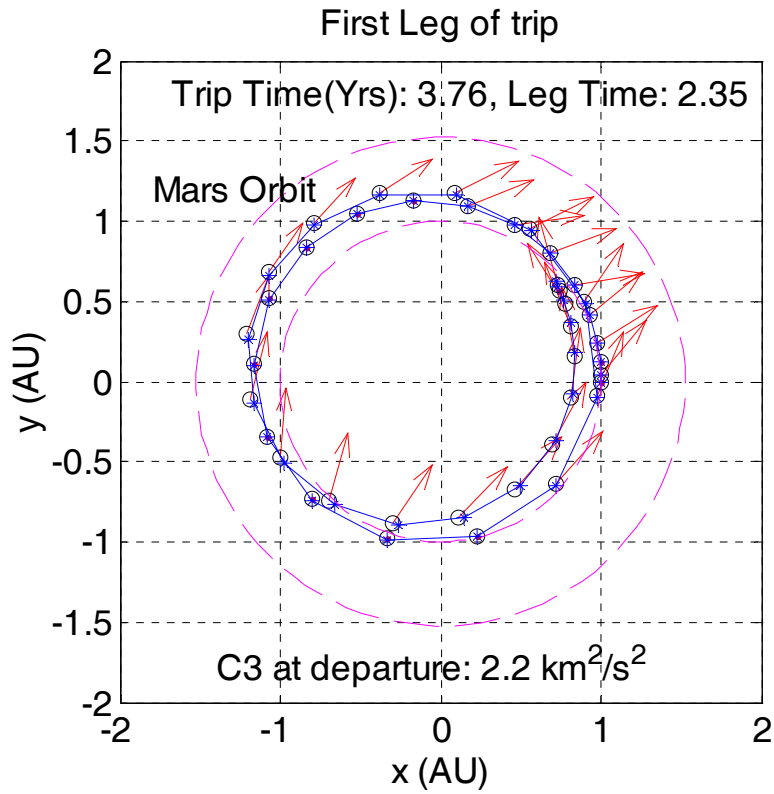
Figure 59   Rendezvous and Return Maximize *e,* Earth to Asteroid Plot (Result 8)

Second Leg of trip

Trip Time(Yrs): 3.76, Leg Time: 1.41
Max Radius: 1.39
Days at NEA: 185

Earth Orbit

Vinf at arrival: 5 km/s

Figure 60   Rendezvous and Return Maximize *e,* Asteroid to Earth Plot (Result 8)

Objective	Semi-major Axis, a (AU)	Eccentricity, e
Maximize a (Result 6)	1.29	0.115
Minimize a (Result 7)	0.86	0.169
Maximize e (Result 8)	1.10	0.267
Minimize e (trivial case)	1	0
Min e & Max a to Min a[8]	1.24 to 0.88	0
Mix a/e ($\alpha = 0.5,\ \beta = -0.5$)[8]	1.02	0.233

Table 5.   Rendezvous and Return Results Summary

2.   Rendezvous and Return Reachable Set

All the results from Table 5 are plotted on an *a-e* two-dimensional projection in Figure 61. However, even though straight lines are drawn between points of this ill-defined boundary, no definitive conclusions can be drawn from targets that lie between points. As demonstrated in the previous chapter, the inner approximation can be constructed using the four extremal cases to show a set of target asteroids that are proven to be within the reachable set (for this level of fidelity). Additionally, using the same four data points, the outer approximation can be constructed to eliminate those target asteroids that are not reachable and proven to be outside the reachable set.
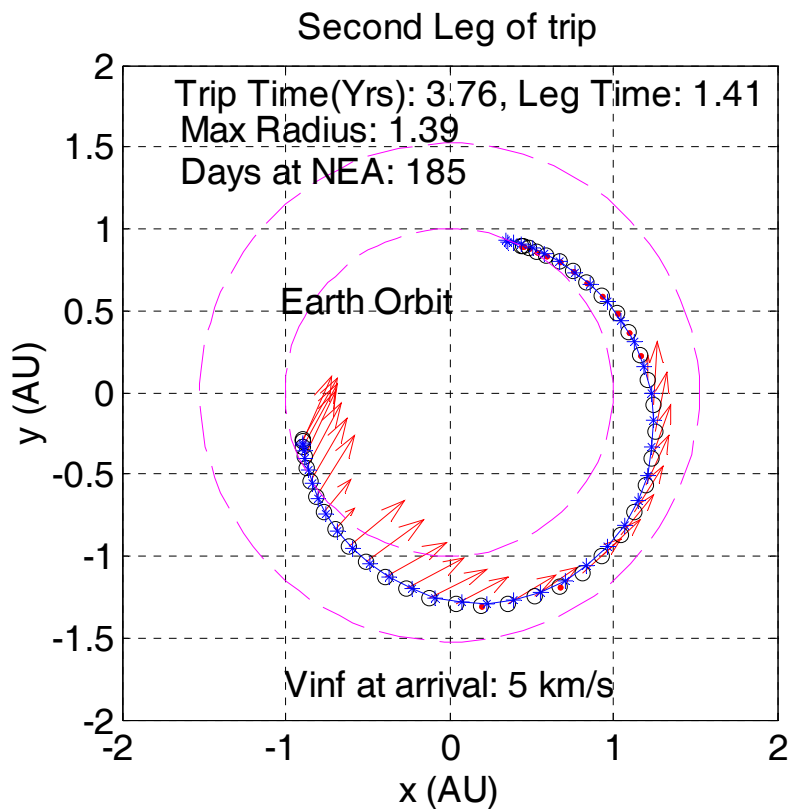
The outer approximation in Figure 63 is much small than that in the asteroid rendezvous scenarios with no return requirement. If every asteroid that is estimated to be large enough to rendezvous (absolute magnitude < 21) were plotted vice only a sample population, there would be 97 asteroids within the outer approximation. This is a drastic reduction from the previously mentioned 6157 target set without using outer approximations. If a guess that a 10° inclination is too large to be achievable, then that 6157 target set is reduced to just 637. Of those, the outer approximation limits the feasible target set to just 23.

---

8 Results not shown here
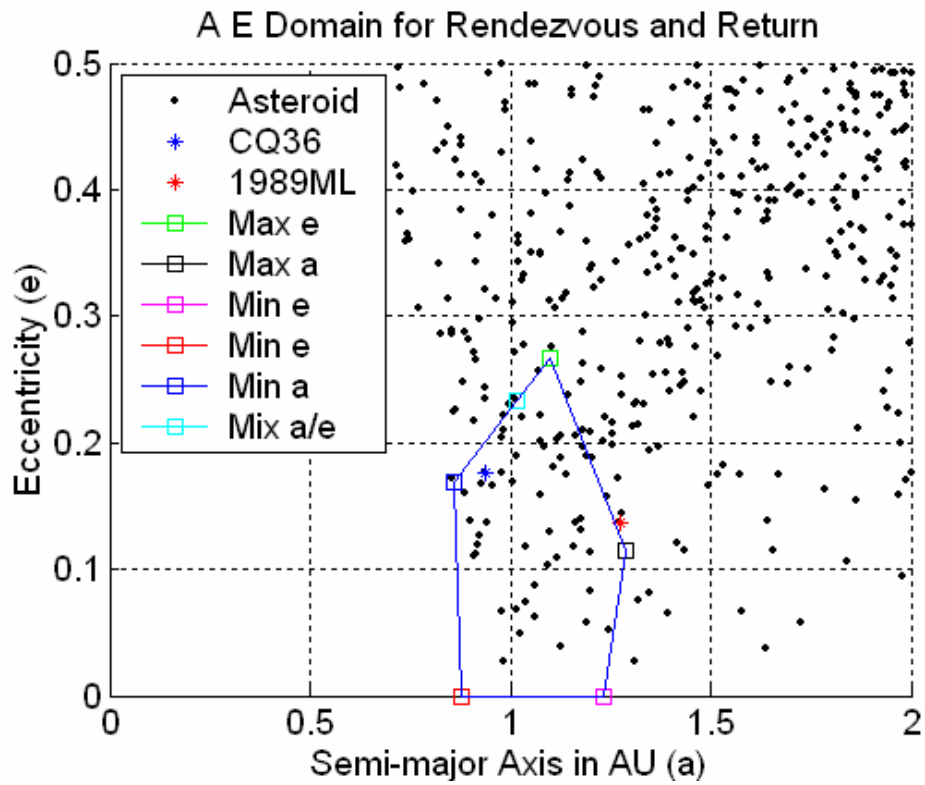
Figure 61   Expanded Detail on Rendezvous and Return Reachable Set Boundary
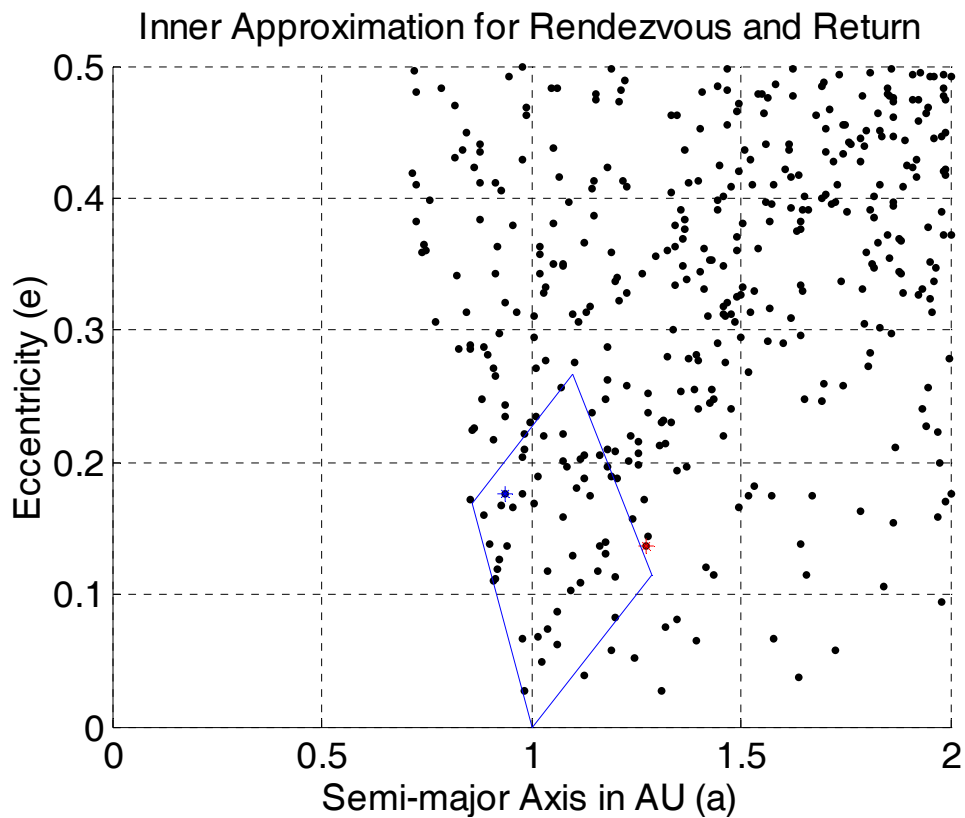
Figure 62   Rendezvous and Return Inner Approximation
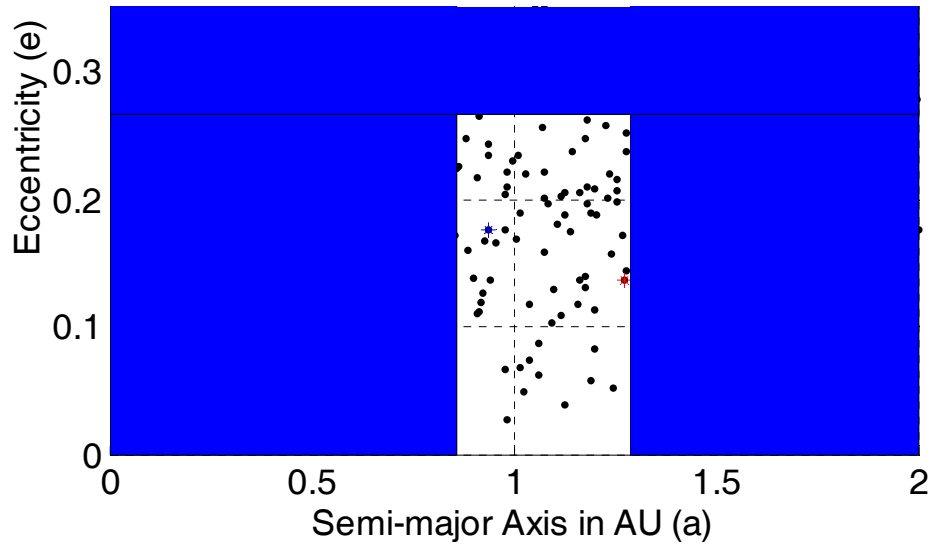
Figure 63   Rendezvous and Return Outer Approximation

Using the extremal solutions that constructed Figure 63, the list of all numbered and unnumbered asteroids (as of August 2004) was screened. Also included was a limit on inclination to be under 10 degrees and the absolute magnitude (*H*) had to be less than 21, which ensures a size of at least 170 meters in diameter for the worst case albedo assumption of 0.25. Table 6 lists the 23 remaining candidates for targets.

Name	*a*	*e*	*i*	ω	Node	MeanAnom	H
1989 ML	1.272	0.137	4.378	183.299	104.416	27.921	19.500
1989 UQ	0.915	0.265	1.291	14.920	178.369	277.578	19.280
1993 HA	1.278	0.144	7.725	263.632	183.388	256.222	19.960
1991 JW	1.038	0.118	8.721	301.858	54.043	217.506	19.280
1997 XR2	1.077	0.201	7.171	84.628	250.888	226.105	20.810
1999 AQ10	0.937	0.234	6.561	299.485	327.414	27.093	20.280
1999 JU3	1.189	0.190	5.884	211.293	251.712	315.076	19.230
1999 RQ36	1.129	0.205	6.024	65.730	2.147	151.450	20.850
2000 EA14	1.117	0.203	3.554	206.019	204.005	173.612	20.900
2000 OK8	0.985	0.221	9.985	166.101	304.653	23.461	19.850
2000 QK130	1.181	0.262	4.720	66.252	174.036	278.922	20.570
2001 CC21	1.032	0.219	4.808	179.077	75.783	184.246	18.390
2001 QC34	1.127	0.187	6.235	215.012	271.935	200.040	19.720
2001 SW169	1.249	0.052	3.555	284.792	8.511	56.585	18.730
2001 TE2	1.084	0.197	7.610	35.683	171.315	333.692	19.870
2002 AW	1.070	0.256	0.567	118.021	162.980	296.191	20.520
2002 CD	0.980	0.177	6.887	331.595	8.877	115.237	20.220
2002 DU3	1.145	0.238	8.703	245.456	0.778	272.307	20.640
2002 OA22	0.936	0.243	6.906	318.276	174.419	197.070	19.300
2002 TD60	1.202	0.083	7.412	343.735	62.719	310.740	19.240
2003 WR21	1.119	0.262	9.276	107.871	85.946	79.601	19.530
2003 YX1	0.879	0.267	5.756	222.828	90.013	115.861	20.830
2004 FM17	0.886	0.249	6.763	196.117	170.140	94.555	19.200

Table 6.   Target Asteroids within the Outer Approximation

VI.   MULTIPLE SAMPLE RETURN

A.   RENDEZVOUS, RETURN TO EARTH, AND REPEAT

The logical extension of the previous problem is to add another rendezvous and return flight following the first sample drop off. The Earth flyby would also be used as a gravity assist opportunity to rendezvous with a completely different target asteroid. Figure 64 and Figure 65 shows the two asteroid rendezvous with all spacecraft trajectories represented by the solid blue lines. Solid black lines show where the spacecraft is rendezvoused with the target and collecting the sample. The dotted lines represent the orbits of the Earth and target asteroids. The difficulty in plotting many revolution orbits is evident, so in this thesis chapter the trajectories are broken up into legs to and from asteroids. However, it will always be true that all legs were simultaneously optimized to extremize the respective objective function.
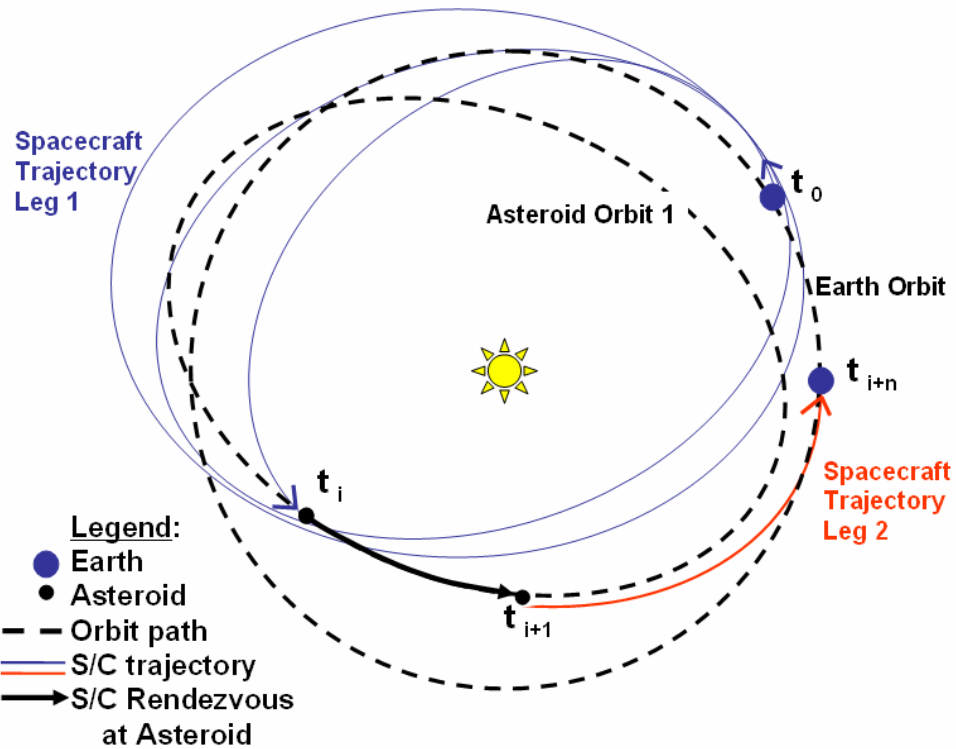
Figure 64   First Asteroid Rendezvous and Return Representation
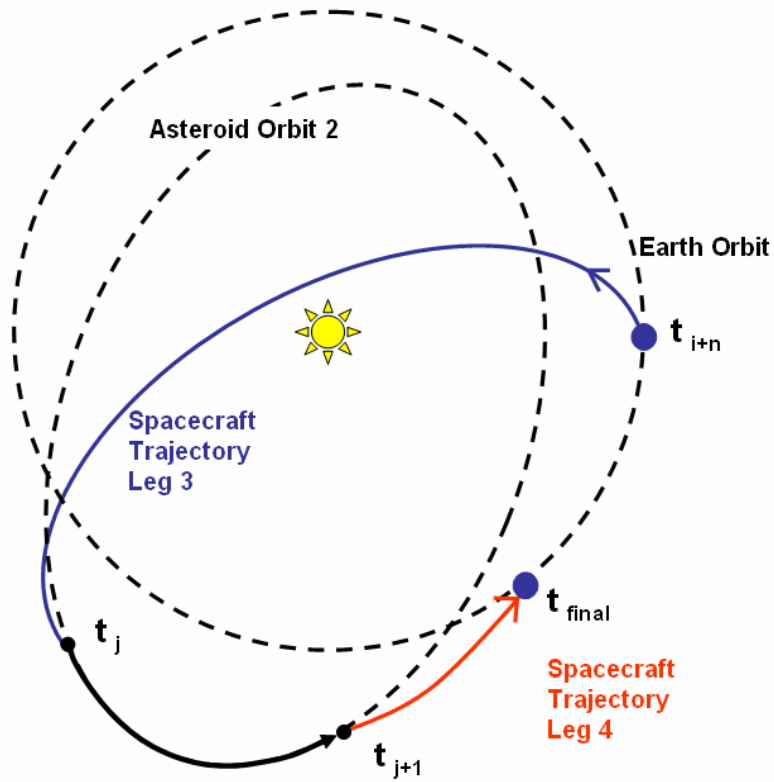
Figure 65   Second Asteroid Rendezvous and Return Representation

1.   Problem Formulation

This problem is very difficult to produce usable results for several reasons involving the complexity, formulation, and cost function. To define constraints at the beginning, first asteroid, Earth return, second asteroid, and final Earth return, a total of 5 knots must be used. The nodes (time) at which these knots occur is defined as follows:

Event Description	Knot Number	Time Notation
Depart Earth	1	t_0
Rendezvous with First Asteroid	2	t_i
Depart First Asteroid	2	t_{i+1}
Flyby/Gravity Assist Earth (first)	3	t_{i+n}
Rendezvous with First Asteroid	4	t_j
Depart Second Asteroid	4	t_{j+1}
Flyby Earth (second)	5	t_f

Table 7. Time Notation for Multiple Sample Return Missions

Accordingly the event constraints grow to over 28 conditions, up from 14 in the previous chapter. Equation 84 lists these, however there are no new types or formulation of constraints are used, only multiple copies to ensure all rendezvous and flyby's are achieved and that states are continuous across the knots. Three important missing features in these event constraints should be noted. First, the stay time at each asteroid is no longer optimized over a range of days. Second, a gravity assist maneuver at the first Earth flyby (t_{i+n}) is not included. This can be constructed in four additional events; much like Rob Stevens did in his thesis [Ref. 4]. Third, the drop mass for the sample recovery vehicle to be returned at first Earth flyby is not included.

Using an inequality constraint for the stay time at each asteroid would never produce feasible results, thus only an equality constraint of 120 days was used. Manually changing what this stay time is set to and rerunning the code could overcome this difficulty.

Similarly, this optimization routine in DIDO never provided feasible results when the four additional constraints to add Earth gravity assist maneuvers to this problem were implemented. Using the same code that Rob Stevens developed and verified [Ref. 4], the current formulation became unstable. Thus, results in this section will not benefit from

the gravity assist maneuver and consequently the second rendezvous points in any results will be less than truly optimal. However, since inclination changes are also not included in this code, the two effects offset each other to some immeasurable degree.

Simulating the drop mass for the part of the spacecraft assigned to reenter Earth's atmosphere and land with the first sample was not implemented. This additional constraint precluded getting feasible results. Including the Earth drop mass would more accurately model the increased force the engine would impart on a lighter vehicle during the second rendezvous and return legs. However, this is also offset due to the lack of modeling real I_{sp} effects when the efficiency of the engine becomes less when the spacecraft is greater than 1.2 AU. Since this usually occurs later in the trajectories or after the actual drop of the sample return vehicle takes place the two inaccuracies would offset each other.

$$\mathbf{e}_{[l,u]} = \begin{bmatrix} r_0 \\ \theta_0 \\ m_0 \\ \left(v_{t_0}-1\right)^2 + v_{r_0}^2 - C_3 \\ r_i - r_{i+1} \\ v_{r_i} - v_{r_{i+1}} \\ v_{t_i} - v_{t_{i+1}} \\ \left(\nu_{i+1} - \nu_i\right)/n_a \\ m_i - m_{i+1} - 20\ kg \\ e_i \\ r_{i+n} - r_{i+n+1} \\ v_{r_{i+n}} - v_{r_{i+n+1}} \\ v_{t_{i+n}} - v_{t_{i+n+1}} \\ \nu_{i+n+1} - \nu_{i+n} \\ m_{i+n} - m_{i+n+1} \\ r_{i+n} \\ \nu_{i+n} - (t_{i+n} - t_0)/n_e \\ \left(v_{t_{i+n}} - \sqrt{\mu_e/r_{i+n}}\right)^2 + v_{r_{i+n}}^2 \\ r_j - r_{j+1} \\ v_{r_j} - v_{r_{j+1}} \\ v_{t_j} - v_{t_{j+1}} \\ \left(\nu_{j+1} - \nu_j\right)/n_a \\ m_j - m_{j+1} - 20\ kg \\ e_j \\ r_f \\ \nu_f - (t_f - t_0)/n_e \\ \left(v_{t_f} - \sqrt{\mu_e/r_f}\right)^2 + v_{r_f}^2 \\ m_f \end{bmatrix} = \begin{bmatrix} [1,1] \\ [0,0] \\ f(C_3) \\ [0,0] \\ [0,0] \\ [0,0] \\ [0,0] \\ [120,120\ days] \\ [0,0] \\ [0,1] \\ [0,0] \\ [0,0] \\ [0,0] \\ [0,0] \\ [0,0] \\ [1,1] \\ [0,0] \\ [0,25\ km^2/s^2] \\ [0,0] \\ [0,0] \\ [0,0] \\ [120,120\ days] \\ [0,0] \\ [0,1] \\ [1,1] \\ [0,0] \\ [0,25\ km^2/s^2] \\ [m_{dry}, m_{dry}] \end{bmatrix} \tag{84}$$

This rendezvous, return and repeat scenario optimization is also complicated due to the cost function, which is more onerous than previous ones. The two weights used to provide four extremal solutions in single asteroid missions must now be increased to four (six for a three-dimensional case) and those can not as easily be manipulated.

$$J = \alpha_1 a(t_i) + \beta_1 e(t_i) + \alpha_2 a(t_j) + \beta_2 e(t_j)$$

(85)

The subscript on the weights denotes the first or second rendezvous. Eight solutions can provide extremal cases of each term in that equation; however this may have less meaning than in previous chapters. For instance, if the maximum a for the second rendezvous is desired and the weights used should be $\alpha_1 = 0$, $\beta_1 = 0$, $\alpha_2 = -1$, and $\beta_2 = 0$, but then the optimal result may end up where there are no asteroids in the first reachable set in which to rendezvous ($a \approx 1$, $e \approx 0$). Similarly, if the maximum a for the first rendezvous is desired and the alpha weights are reversed, then commonly seen in this DIDO implementation is that it will find a solution where an Earth flyby event occurs almost exactly at the same point as a Rendezvous. Thus, the last leg of the journey requires little to no fuel. The chances of that geometry occurring in reality are very low.

As seen in the "Maximize Eccentricity" case for the single asteroid sample return missions, the weights normally set to zero to achieve extremal cases must have a slight non-zero value in order to produce feasible and optimal results. The most consistent and repeatable runs with feasible and optimal results generally occur when four non-zero weights are used for this formulation. Unfortunately, using non-zero values precludes using the concepts of inner and outer approximations to explore the reachable set of asteroids since it is predicated on extremal solutions of the problem. Whenever possible, very low values of weights on the order of $1x10^{-5}$ are used to minimize the effect on the optimal solution.

A cost function results in a simple scalar number to judge the performance of one solution against another. The sensitivity to its formulation may be partly due to the fact that Equation (85) involves non-linear and complex transformations from the state information. Indeed, when this cost function was simplified to be a linear function of the

radial state such as in Equation (86), feasible and optimal results were much easier obtained and in also allowed more detail to be added to the optimization, such as optimizing a range of stay times.

$$J = \pm r(t_i) \text{ or } J = \pm r(t_j) \tag{86}$$

Lastly, difficulty with achieving feasible or optimal solutions where the dynamics, constraints, and bounds have show satisfactory performance in simpler formulations may be due to a poor quality of initial guesses. In this scenario, a simple 5 point guess of states, controls, and times based off no previous results was used. This guess is used to start this 5 knot optimization using 12 nodes between each knot (total of 48). This lower order discretization is then bootstrapped into a better state-control-time guess structure to start a high order discretization of 120 nodes, or 30 nodes between each knot. The simple and easy "no guess" bootstrapping method described previously for all trajectory optimizations may not be the best way to formulate this problem.

2.   Solutions

Without a basic gravity assist maneuver modeled, any results can not be rigorously taken as limiting the number of feasible targets for a multiple asteroid mission. Even though some of the gravity assist ΔV would be used to change inclination, a portion might be used to enhance the planar performance of the spacecraft. Thus, unlike the 2 DOF Rendezvous and Return solutions in the previous chapter, some margin would likely be needed to the any of the "optimal" results present hereafter. Additionally, the largest challenge to obtain feasible and optimal results with this problem formulation is choosing the weights used on the cost function. As seen with the maximize eccentricity case of the previous chapter, non-zero weights are sometimes required to achieve usable results, so any usable results using the cost function in Equation (85) are only "nearly extremal".

Figure 66 shows the result of one problematic optimization result where both $a(t_j)$ and $e(t_j)$ are maximized (equal negative weights on each and near zero weights on $a(t_i)$ and $e(t_i)$). This plot depicts the S/C originating at Earth ($a = 1$, $e = 0$), making its first rendezvous at the point ($a = 1.2$, $e = 0.04$), intercepting Earth to drop off the sample at the point ($a = 1.18$, $e = 0.15$), then rendezvousing with another asteroid at ($a =$

1.15, *e* = 0.12), and just moments later and expended almost no fuel to get back to Earth for the final flyby. You cannot see the second rendezvous in Figure 66 since it is at the same point as the second Earth flyby. It is unlikely that such a fortuitous astrodynamical event will occur where a reachable asteroid is nearly crossing Earth's orbit just following the second sample collection. Thus, these types' of results may not be very realistic and have been excluded from the rest of the results shown.
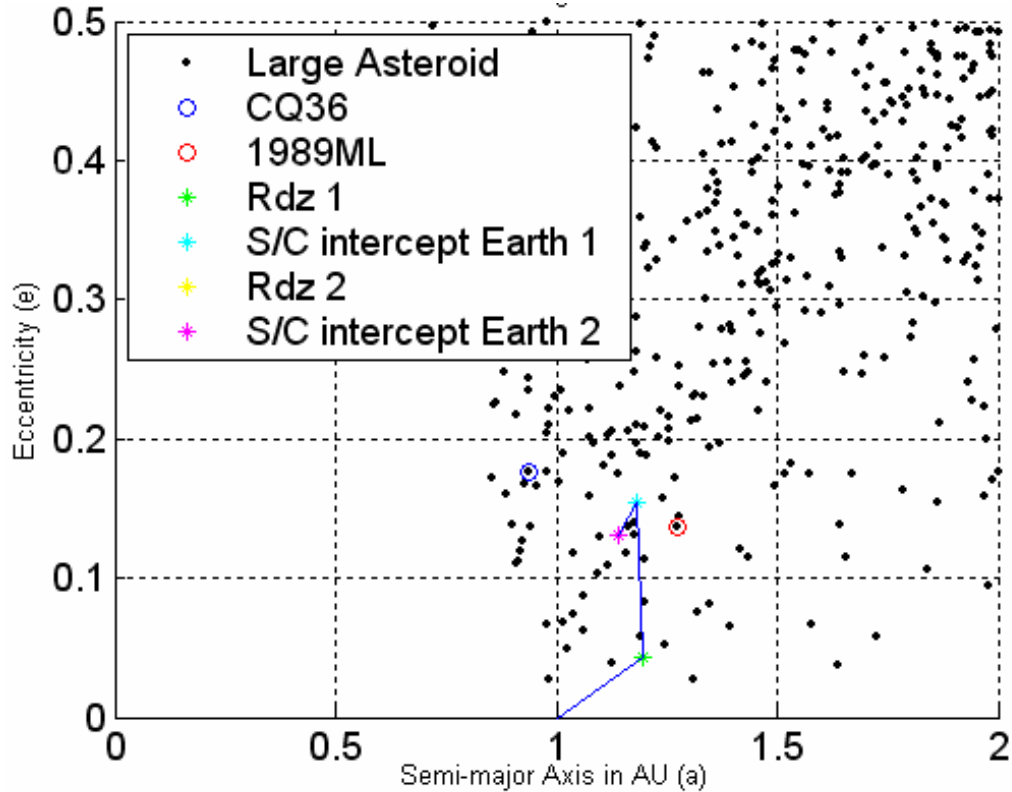

Figure 66   Multiple Asteroid Rendezvous and Sample Return *a-e* Domain Result[9]

a.   Maximize Second Rendezvous Semi-major Axis

Figure 67, Figure 68 and Figure 69 shows the state and control history for a locally optimal solution when maximizing the semi-major axis, *a*, of the second asteroid rendezvous orbit. The control history was propagated with ODE23tb and has a reasonably small error from the DIDO solution as illustrated by the how near the discrete DIDO points (circles) match the continuous (line) representing propagated result of the optimal control history. Also since no constraints were broken in this solution, it proves

---

[9] As mentioned previously the S/C Intercept Earth 1 point is plotted over the Rdz 2 point.

feasible. Since the original solution from the formulation previously described had enough error to put feasibility in doubt, the original solution was "bootstrapped" into a guess of a new DIDO run with twice the number of nodes to converge the two state plots and eliminate most of the error. Thus, the number of node points solved for and illustrated in this solution is 240, vice the 120 mentioned in formulating the problem. It took an additionally 13 runtime minutes on the computer, from the initial 6.5, to perform this additional step.
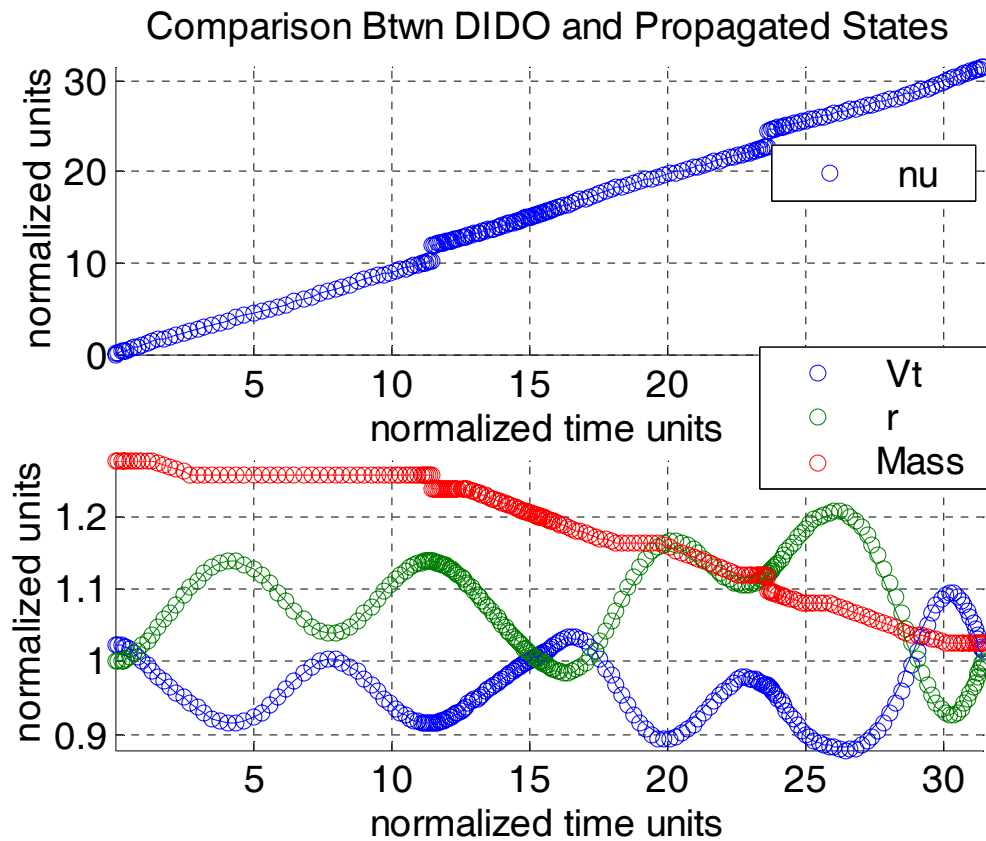

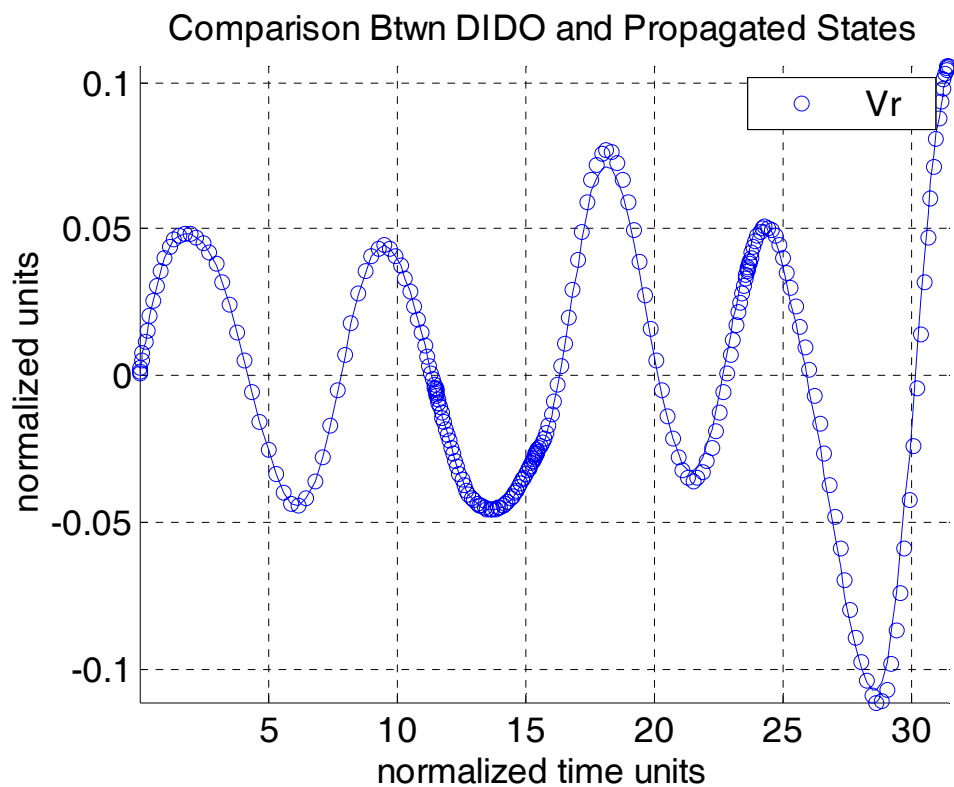Figure 67   Rendezvous, Return & Repeat Maximize *a*, State History

Figure 68   Rendezvous, Return & Repeat Maximize *a*, State History (cont)
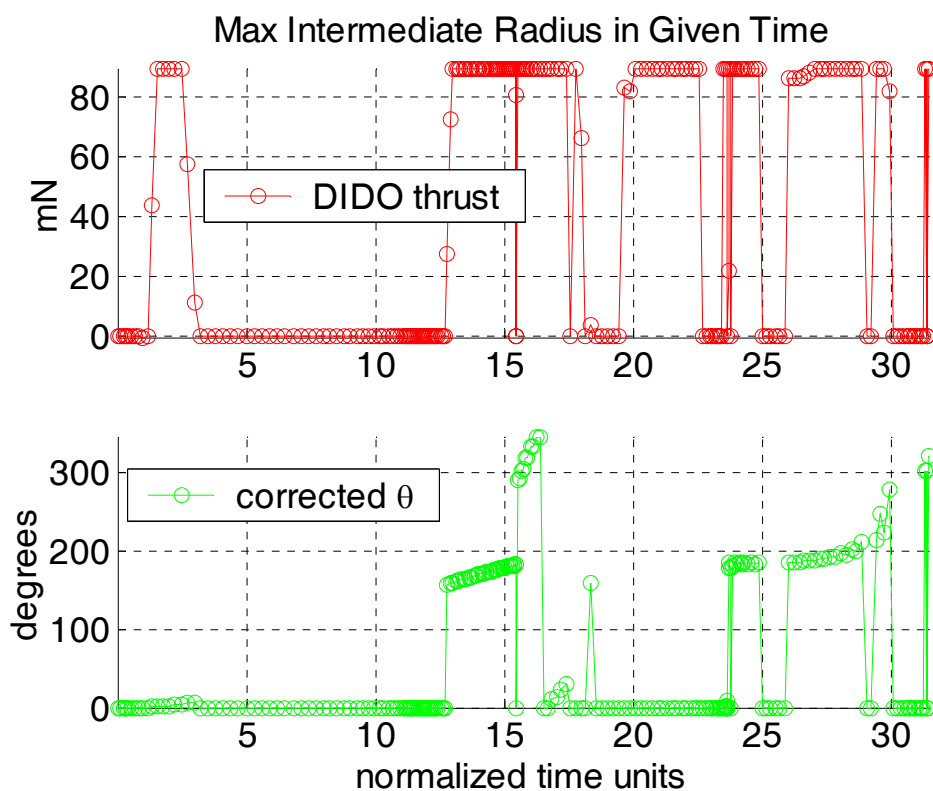

Figure 69   Rendezvous, Return & Repeat Maximize *a*, Control History

For this result the weights used were: $\alpha_1 = -0.001$, $\beta_1 = -0.001$, $\alpha_2 = -1$, $\beta_2 = -0.01$. This would highly emphasize maximizing the second rendezvous a. Figure 70, Figure 71, Figure 72, and Figure 73 plots each leg of the final trajectory separately, even thought the entire flight was optimized simultaneously. The stay times at the asteroid was fixed at 120 days. The final result is summarized in Figure 74. It shows the first asteroid rendezvous orbit at $a = 1.0887$ and $e = 0.0462$ and the second asteroid rendezvous orbit at $a = 1.1817$ and $e = 0.0628$, along with the orbit the spacecraft was in at each Earth flyby. This solution was deemed "probably optimal" by DIDO.

Figure 70   Rendezvous, Return & Repeat Maximize a, Earth to First Asteroid Plot

Second Leg of trip

Trip Time(Yrs): 5.13, Leg Time: 0.64

Mars Orbit

Earth Orbit

Days at NEA: 120

Vinf at arrival: 0.839 km/s

Figure 71   Rendezvous, Return & Repeat Maximize *a*, First Asteroid to Earth Plot

Third Leg of trip

Trip Time(Yrs): 5.13, Leg Time: 1.34

Mars Orbit
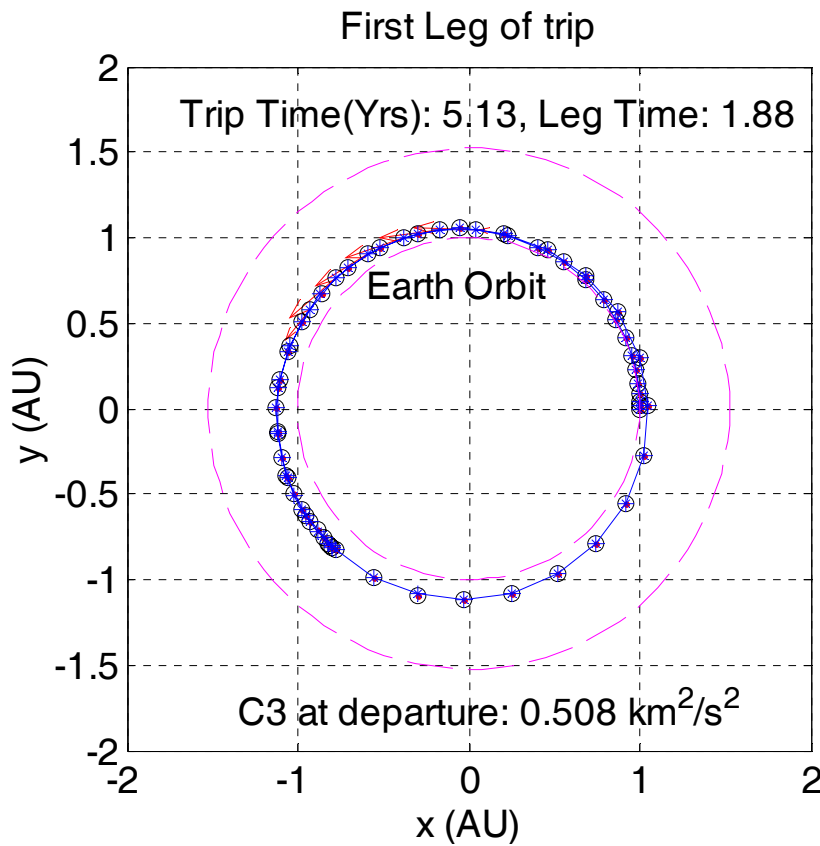
C3 at departure: 0.705 km^2/s^2

Figure 72   Rendezvous, Return & Repeat Maximize *a*, Earth to Second Asteroid Plot

Figure 73   Rendezvous, Return & Repeat Maximize *a*, Second Asteroid to Earth Plot

Figure 74   Rendezvous, Return & Repeat Maximize *a*, *a-e* Domain Plot

b. *Maximize Second Rendezvous Semi-major Axis (refined weights)*

In order to attempt to make the weights used in the previous result more extremal, the entire previous 240 node solution was used as the starting guess for a new optimization run. Non-zero weights were still able to be used, however the following weights did obtain equally feasible and locally optimal results: $\alpha_1 = -0.00001$, $\beta_1 = -0.00001$, $\alpha_2 = -1$, $\beta_2 = -0.00001$. The plots showing the states, controls, and orbits are not shown since they did not discernibly change from the previous solution. The second asteroid rendezvous maximum a did improve marginally from 1.1817 to 1.1836, however, this 0.16% change is arguably within the accuracy of DIDO's solution and the problem formulation fidelity. It is unproven here, but most likely true that the ability to choose an optimal stay time would have more effect on the a-e domain plot than the ability to have absolute zero valued weights.

c. *Maximize Second Rendezvous Eccentricity*

For brevity, the state and control history for a locally optimal solution when maximizing e of the second asteroid rendezvous orbit is not explicitly shown here. The control history was propagated with ODE23t and has a very small error from the DIDO solution as illustrated by the how near the discrete DIDO points (circles) match the continuous (line) representing propagated result of the optimal control history. Also since no constraints were broken in this solution, it proves feasible. Since the original solution from the formulation previously described had enough error to put feasibility in doubt, the original solution was "bootstrapped" into a guess of a new DIDO run with twice the number of nodes to converge the two state plots and eliminate most of the error. Thus, the number of node points solved for and illustrated in this solution is 160, vice the 120 mentioned in formulating the problem.

For this result the weights used were: $\alpha_1 = -0.001$, $\beta_1 = -0.01$, $\alpha_2 = -0.01$, $\beta_2 = -10$. This would highly emphasize maximizing the second rendezvous e. Figure 75, Figure 76, Figure 77, and Figure 78 plots each leg of the final trajectory separately, even thought the entire flight was optimized simultaneously. The stay times at the asteroid was fixed at 120 days. The final result is summarized in Figure 79. It shows the first asteroid rendezvous orbit at $a = 1.1174$ and $e = 0.1154$ and the second

asteroid rendezvous orbit at a = 1.1410 and e = 0.1315, along with the orbit the spacecraft was in at each Earth flyby. DIDO deemed this solution "probably optimal".

Figure 75   Rendezvous, Return & Repeat Maximize e, Earth to First Asteroid Plot
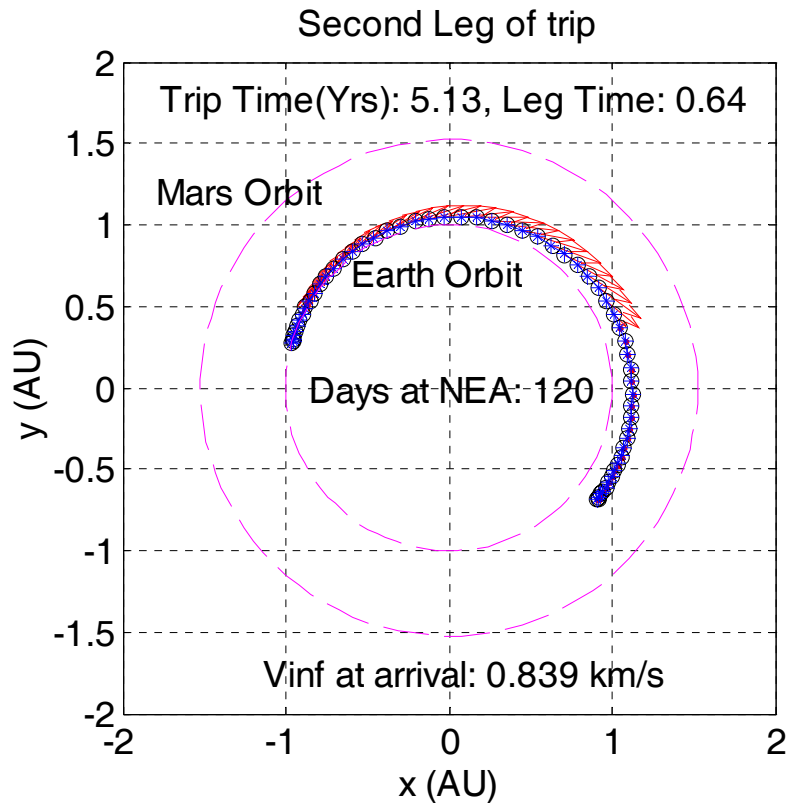
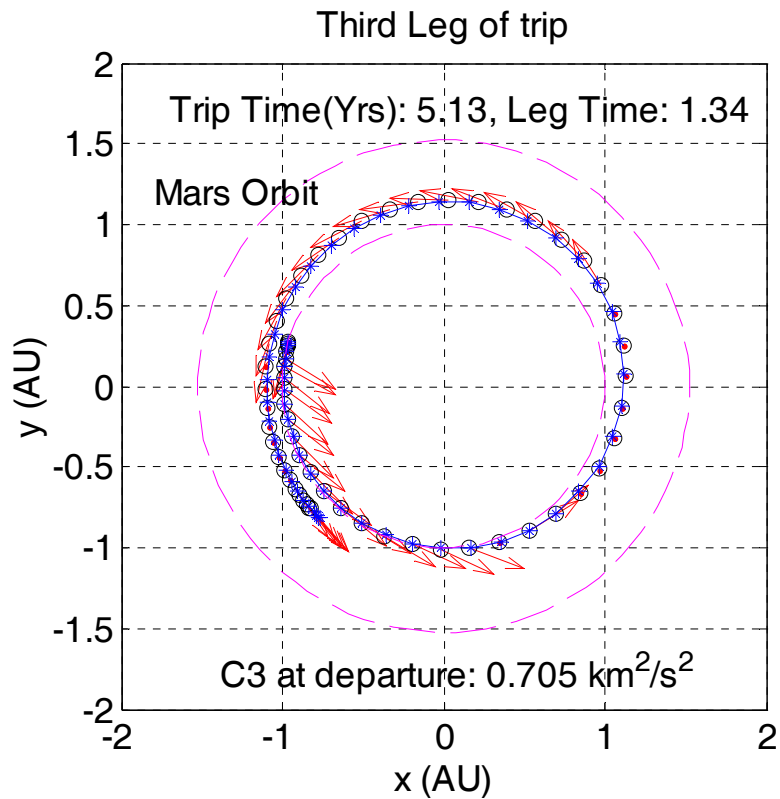Figure 76   Rendezvous, Return & Repeat Maximize *e*, First Asteroid to Earth Plot

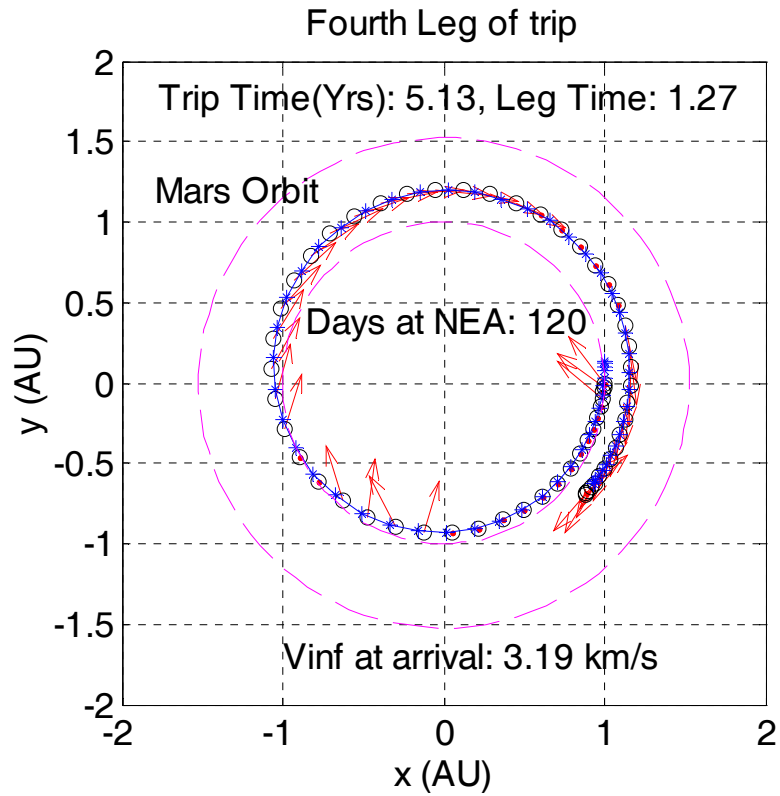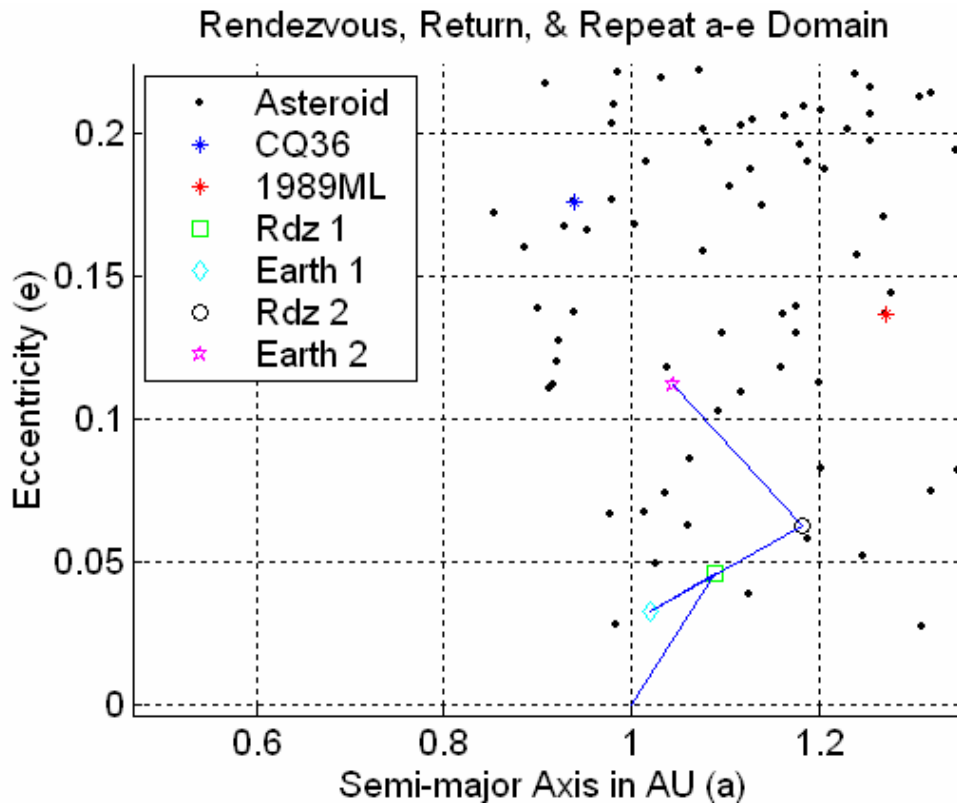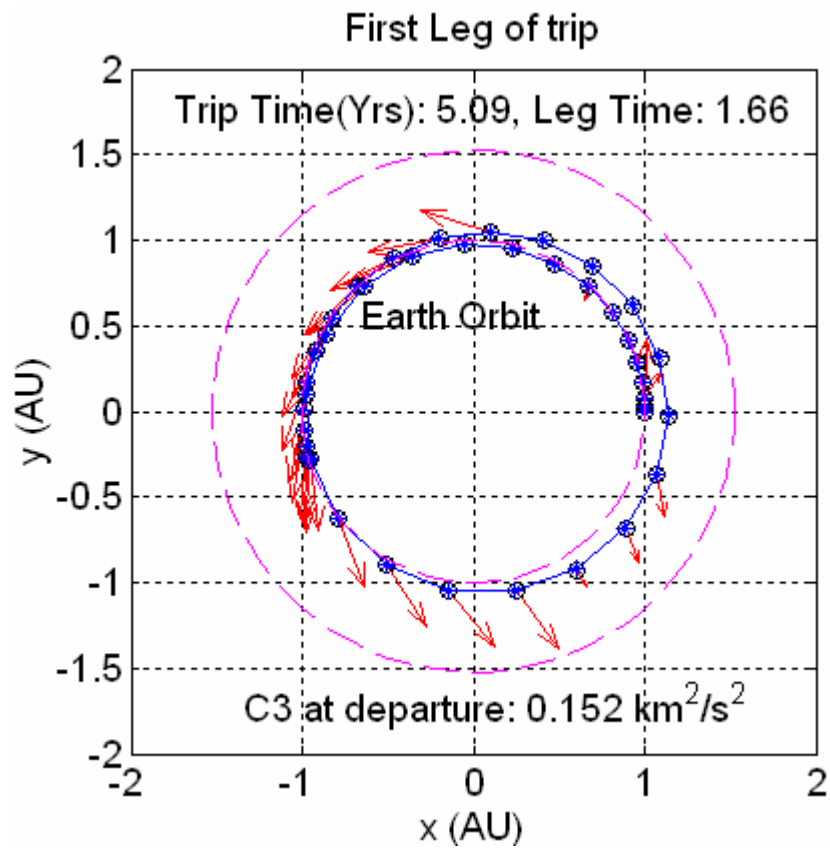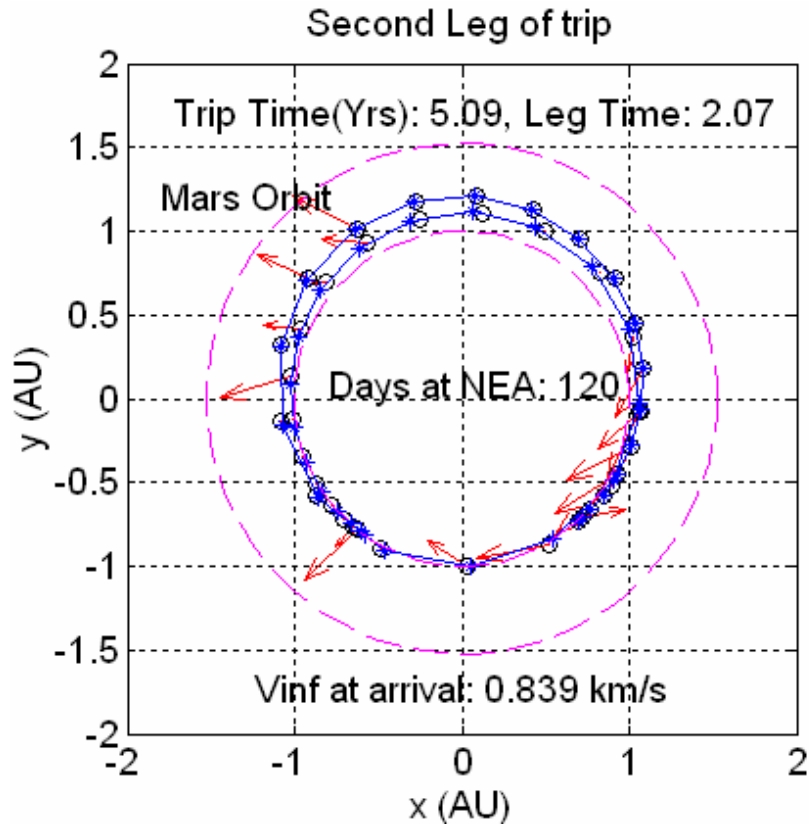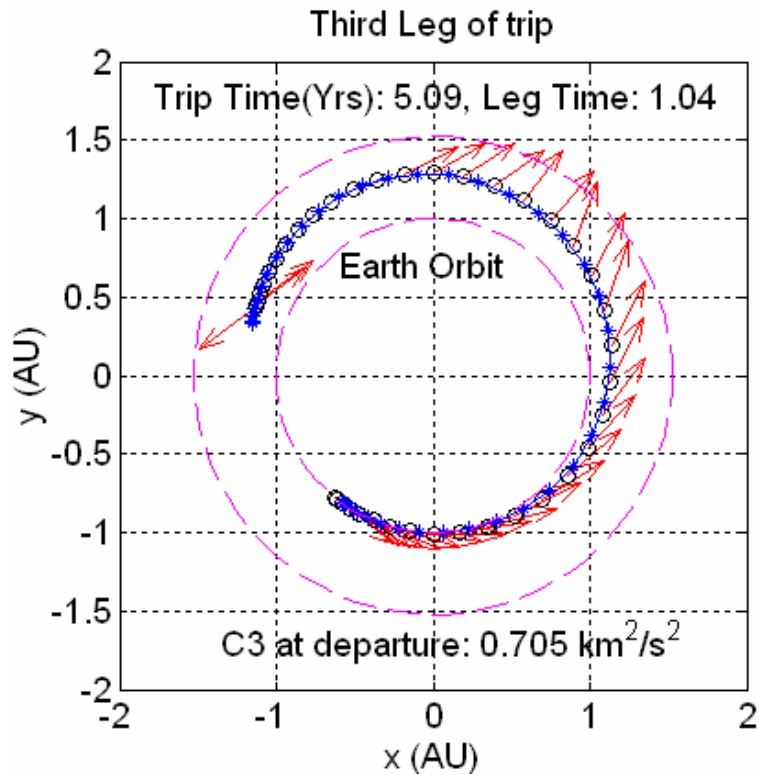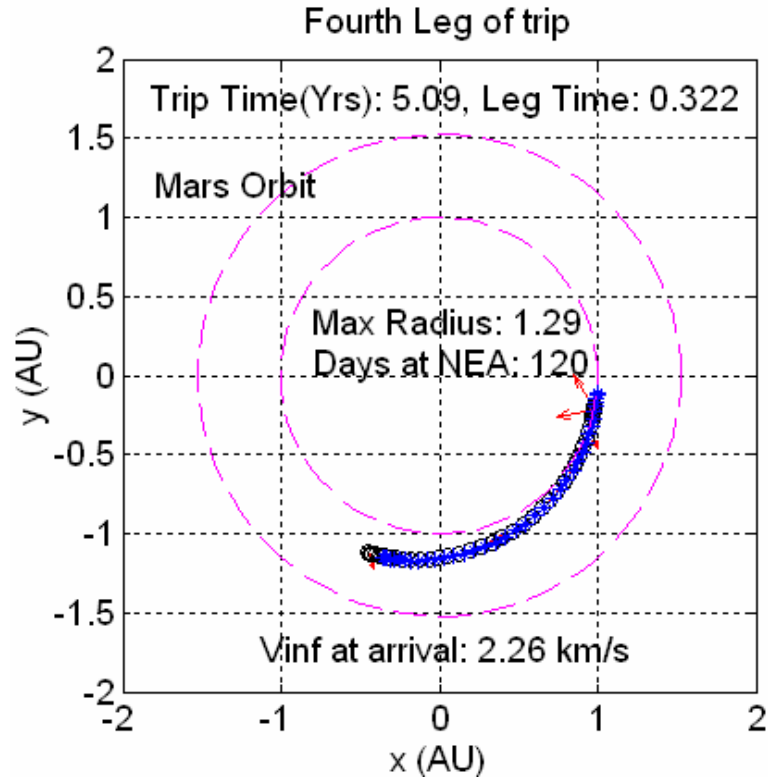Figure 77   Rendezvous, Return & Repeat Maximize *e*, Earth to Second Asteroid Plot

104

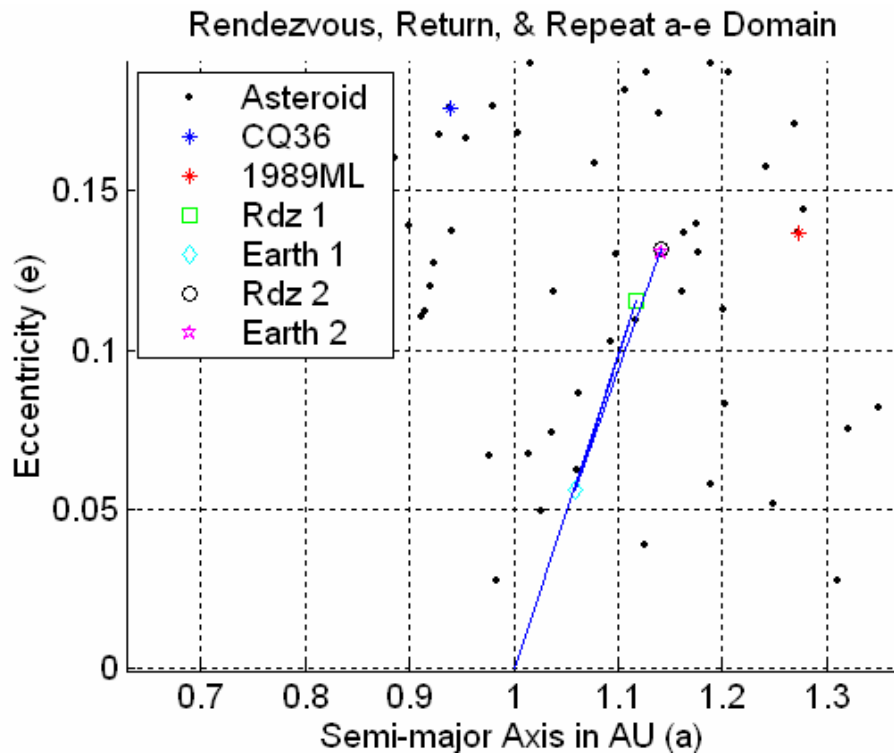Figure 78   Rendezvous, Return & Repeat Maximize *e*, Second Asteroid to Earth Plot

Figure 79   Rendezvous, Return & Repeat Maximize *e*, *a-e* Domain Plot

105

d. *Minimize First Rendezvous Radius*

As previously mentioned, the ability to formulate a linear and simplistic cost function greatly improves the ability to consistently achieve feasibly and optimal results and have less sensitivity to initial guesses and weights. If a cost function is used that is solely dependant on the radius of the spacecraft trajectory, some usefulness may be extracted with much less effort. Since radius from the sun is the first state in this dynamical system, its computation at any rendezvous point is trivial.

The 120 node state and control histories are plotted in Figure 80, Figure 81, and Figure 82. The optimal controls determined by DIDO were propagated with ODE45 with acceptable error. The minimum radius of the first rendezvous orbit was determined to be 0.8895 AU for a 120 day stay time and leaving enough fuel to complete another rendezvous and return mission that had an 120 day stay time. Figure 83, Figure 84, Figure 85, and Figure 86 plots the trajectory of the spacecraft. The *a-e* domain plot in Figure 87 is included for completeness, even though it has little meaning in the minimizing radius case.
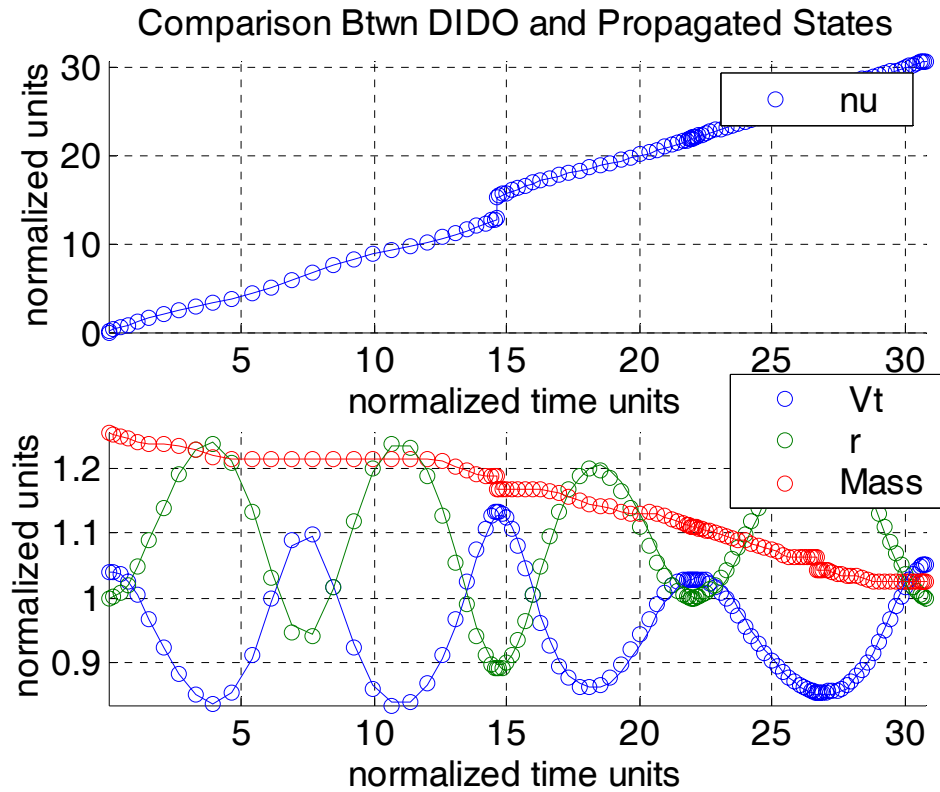
Figure 80   Rendezvous, Return & Repeat Minimize Radius, State History

106
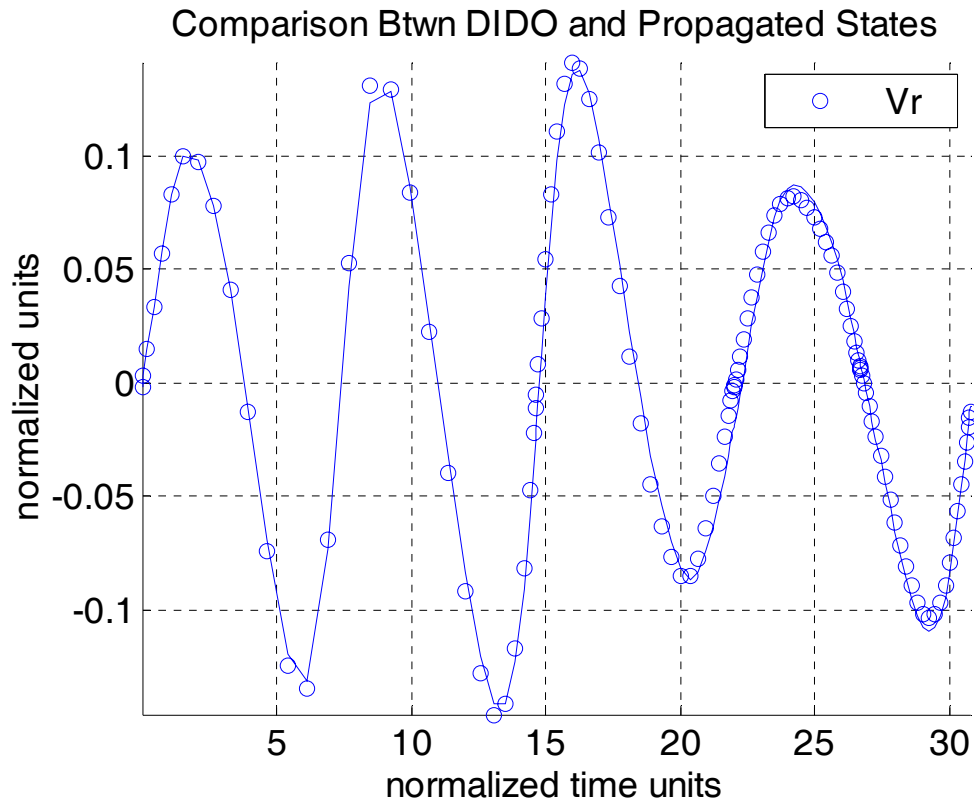
Figure 81   Rendezvous, Return & Repeat Minimize Radius, State History (cont)
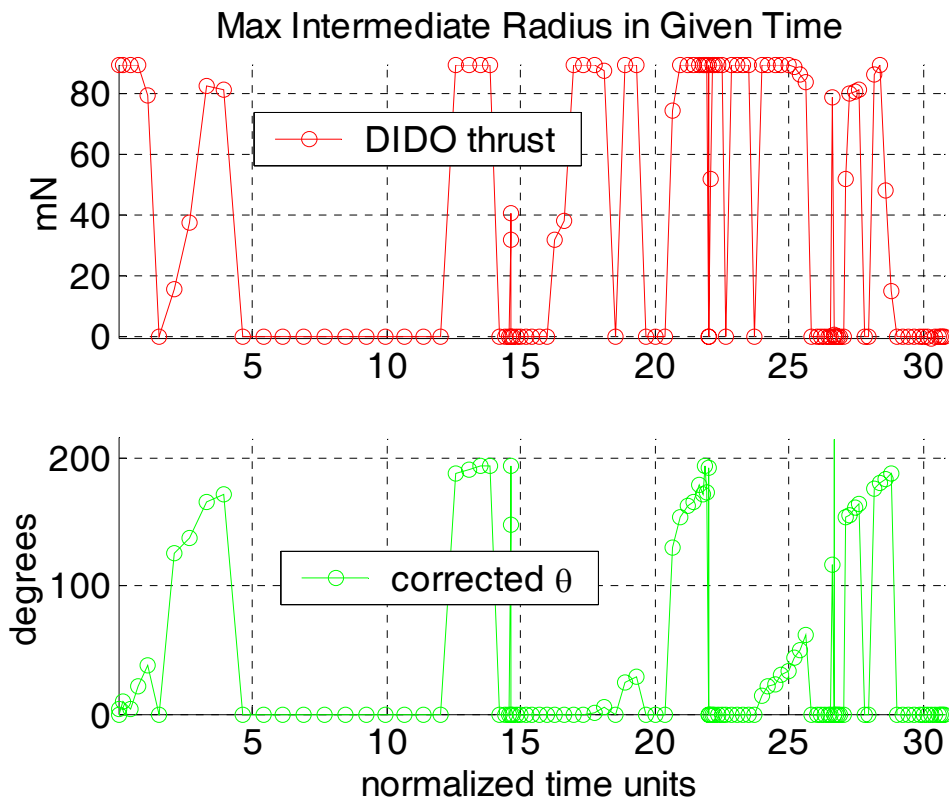
Figure 82   Rendezvous, Return & Repeat Minimize Radius, Control History

First Leg of trip

Trip Time(Yrs): 5.02, Leg Time: 2.38

Mars Orbit

C3 at departure: 1.32 km^2/s^2
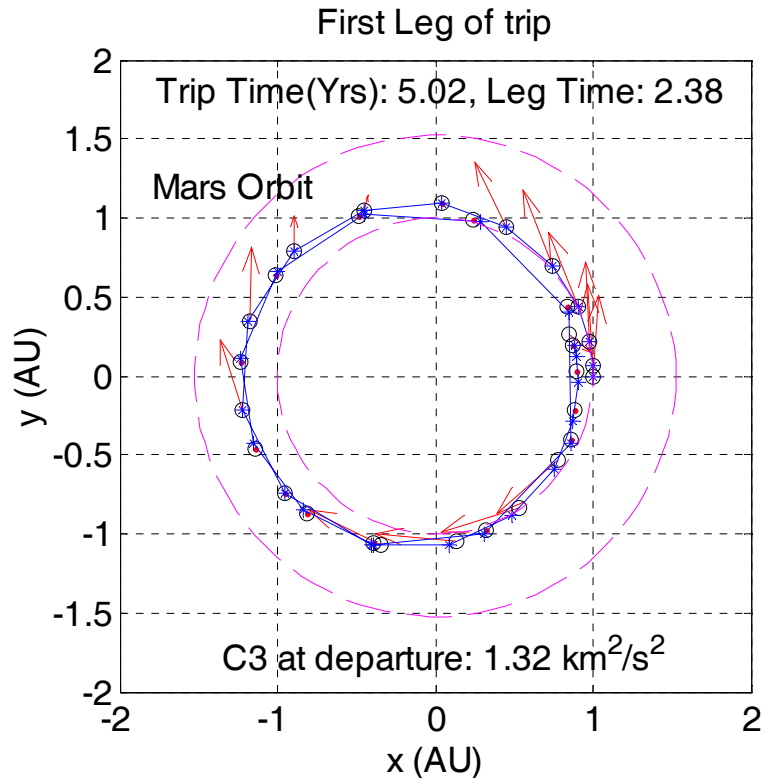
Figure 83   Rendezvous, Return & Repeat Minimize Radius, Earth to First Asteroid Plot

Second Leg of trip

Trip Time(Yrs): 5.02, Leg Time: 1.2

Mars Orbit

Max Radius: 1.19
Days at NEA: 120

Vinf at arrival: 0.839 km/s
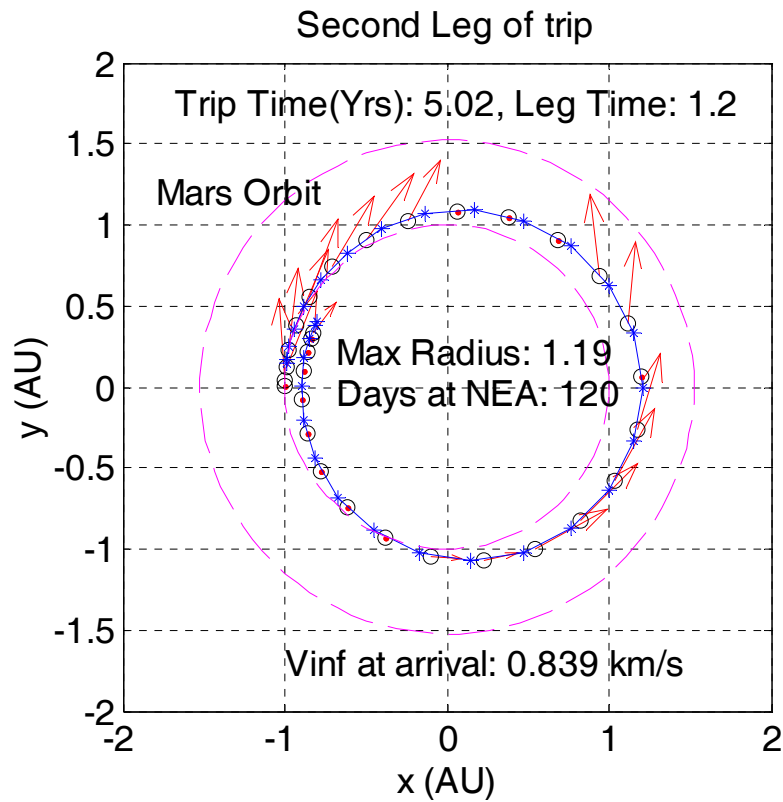
Figure 84   Rendezvous, Return & Repeat Minimize Radius, First Asteroid to Earth Plot
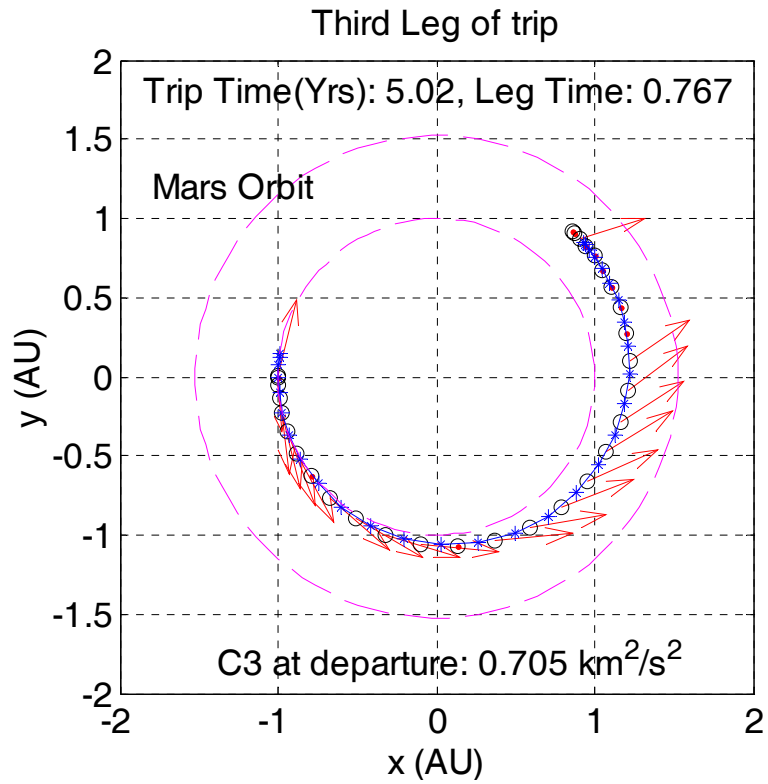
Figure 85   Rendezvous, Return & Repeat Minimize Radius, Earth to Second Asteroid Plot
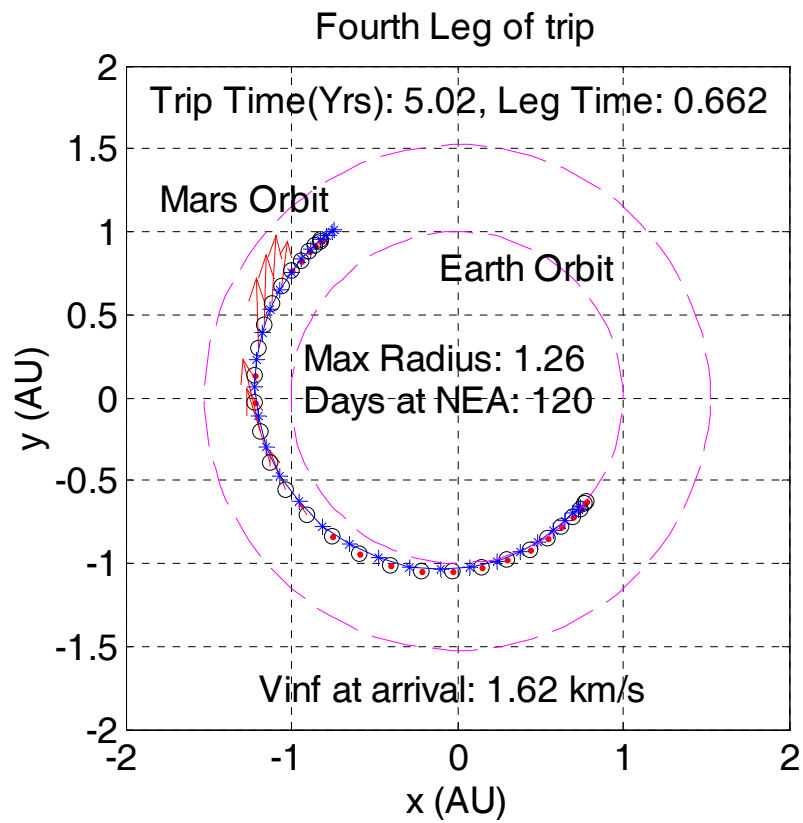
Figure 86   Rendezvous, Return & Repeat Minimize Radius, Second Asteroid to Earth Plot
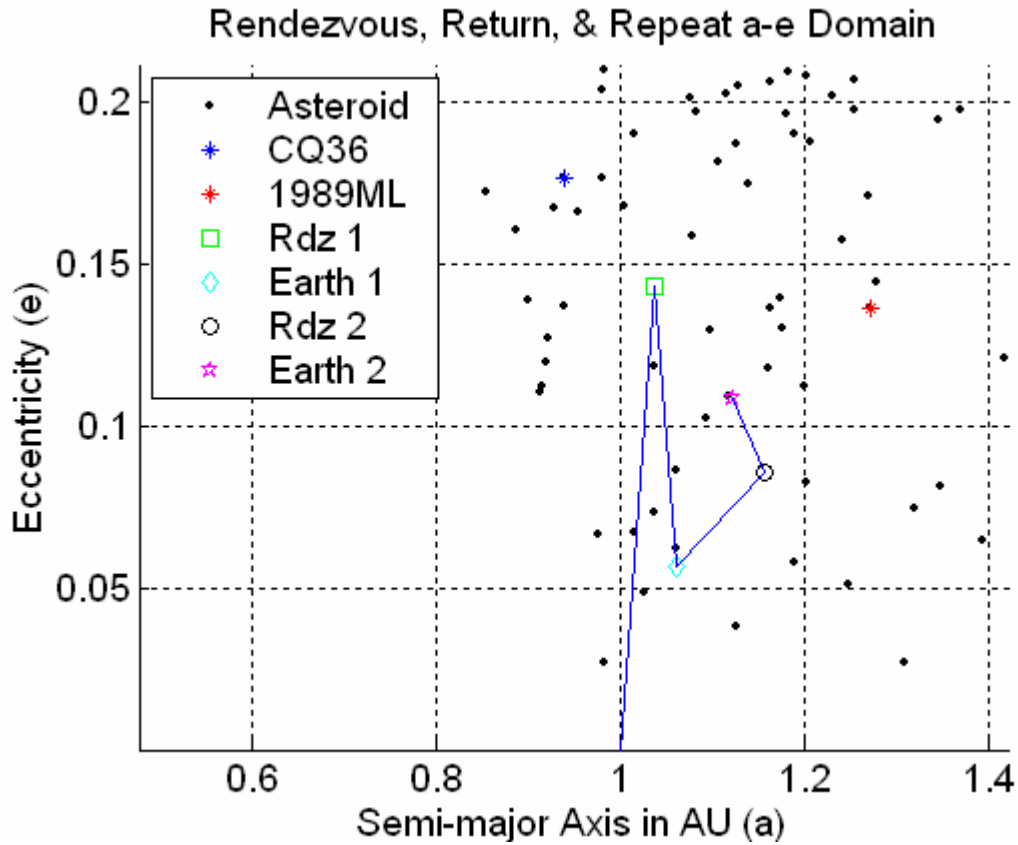
Figure 87   Rendezvous, Return & Repeat Minimize Radius, *a-e* Domain Plot

The improvement in NLP solution performance was significant enough to rerun all possible min/max radius cases with a stay time range between 90-120 days be an optimization parameter. The final results are not shown in this thesis, but captured in Table 8 below.

	Radius (in AU)	Notes:
Minimize First Asteroid Radius	0.8756	
Minimize Second Asteroid Radius	0.9129	
Maximize First Asteroid Radius	1.1568	Result was only "nearly optimal" by DIDO
Maximize Second Asteroid Radius	1.2669	

Table 8.   Rendezvous, Return & Repeat Min/Max Radius Results

110

3.   Rendezvous, Return, and Repeat Reachable Sets

There are some computational difficulties with formulating reachable sets, or their mathematically valid subsets of the inner and outer approximations presented in previous chapters. For the asteroid rendezvous, sample return and repeat trajectories, computing all the optimal-extremal solutions by DIDO with the current formulation has not been demonstrated. Finding four "near extremal" sets of weights that produces feasible and optimal results by refining the weights used through bootstrapping very good guesses into higher order solutions has been shown, but is very time-consuming and not subject to consistent methodologies. However, even spending the effort in tweaking optimization code will still leave other difficulties associated the applicability of solutions with the fidelity of a 2 DOF model missing stay time optimization, missing a gravity assist optimization, and having a low level engine model. With additional effort applied to reformulating the problem, dynamics model, scaling used, and/or coordinate system, these computational difficulties can probably be resolved [Ref.s 4,8]. However, the conceptual difficulties with defining usable subsets of the reachable set still exist.

Figure 66, Figure 74, Figure 79, and Figure 87 all show optimal solutions for some objective and thus these rendezvous points lie on a boundary of the reachable set. A minor conceptual problem is that even if a target lies within the inner approximation found for the maximum a for the second rendezvous, this assumes an optimal first rendezvous as determined by DIDO. Choosing a real asteroid orbit other than the optimal first rendezvous conditions will reduce the maximum a or e that is achievable for the second rendezvous. Thus, inner approximations that should guarantee achievable solutions for specific targets are not valid. As before, outer approximations may be helpful in excluding many of the targets if the computational difficulties are eliminated.

Since a stay time optimization was included, the results from optimizing the minimum and maximum radius will be used to develop reachable sets. If the gravity assist maneuver could be included for the results in Table 8, these can be used to define the reachable set boundaries for the first and second asteroid rendezvous. These approximations would be only one-dimensional since there is only one maximum or minimum radius. If an asteroid's perihelion distance is greater than the maximum radius, then it would lie outside the reachable set. If an asteroid's aphelion distance is less than

the spacecraft trajectory minimums, then they would lie outside the reachable set as well. From [Ref. 14], the perihelion and aphelion distance can be calculated by

$$r_p = a(1-e) \ , \ r_a = a(1+e) \tag{87}$$

and resulting in the functions to find a reachable set on *a-e* domain are

$$r_{\min_{1,2}} \geq a(1-e) \ \cap \ \boldsymbol{r}_{\max_{1,2}} \geq a(1+e) \tag{88}$$

which can be plotted to provide continuous boundaries as in Figure 88.
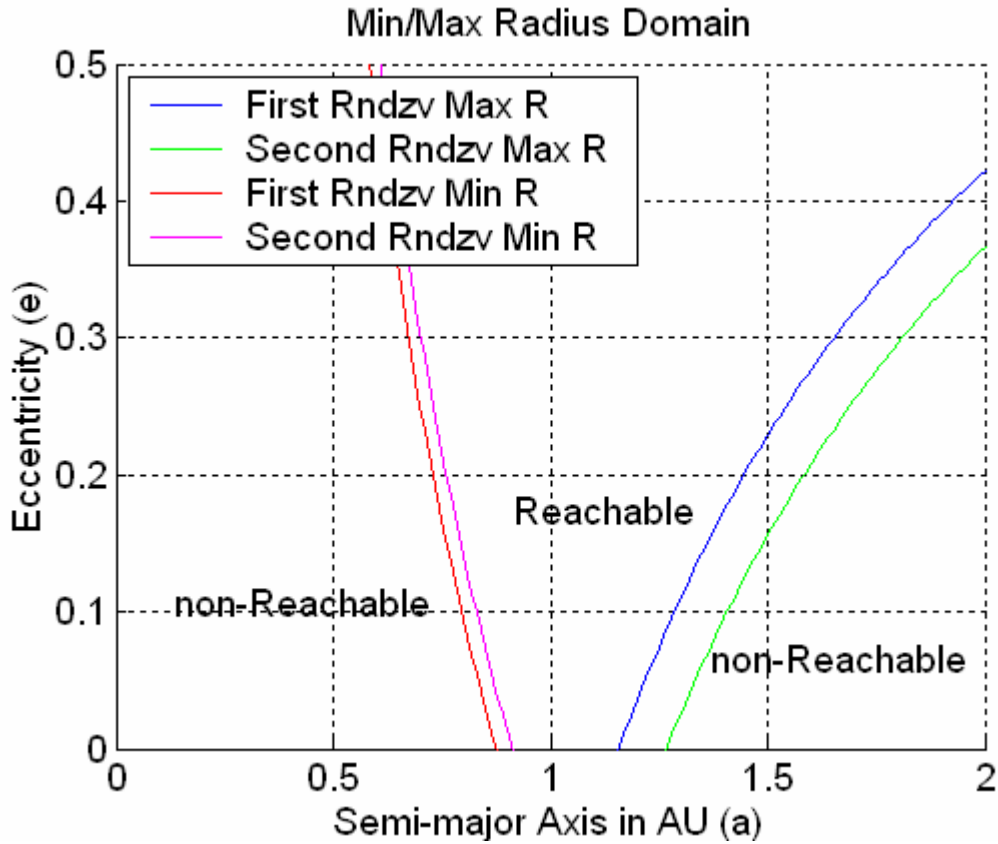
Figure 88   Reachable Boundary Based off Min and Max Radius Results

When the maximum *e* is plotted, then the reachable set is fairly well defined. The maximum first asteroid e is plotted in Figure 89. Certainly, the a-e domain might be more limiting, however without even the stay time optimization, many of the reachable asteroids could be left out. This process can be refined to ensure that any reachable asteroid orbit has at least 90 days within the maximum or minimum radius limits to further exclude targets.

112
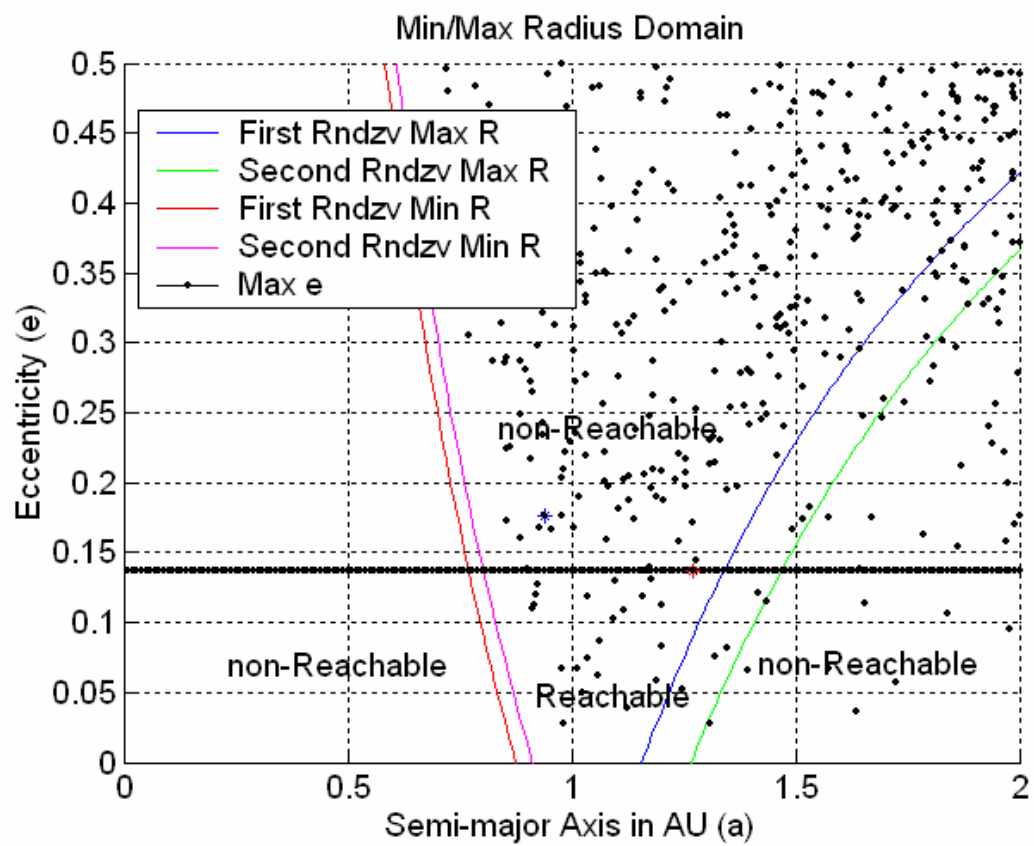
Figure 89   Reachable Set for Two Asteroid Sample Return Missions

113

THIS PAGE INTENTIONALLY LEFT BLANK

VII. CONCLUSIONS AND FUTURE WORK

Theoretically, targets within a reachable set boundary are feasible targets for a sample return mission. For the single sample return missions, this allowed defining outer approximations where the extremal solutions necessarily bounded the number of possible targets. The inner approximation for the single sample mission is less useful since it provides few guarantees about reachablity without a very large number of optimal trajectories plotted to well define a boundary. However, when combined with a few non-extremal points, it can give quickly show a low-order shape that is indicative of the actual reachable set. This conclusion is demonstrated in Table 6 where the feasible targets for a asteroid rendezvous and return mission with any gravity assist maneuvers is limited to just 23 possible candidates for a given theoretical spacecraft, propulsion system, and launch vehicle.

For the multiple sample return mission, the fidelity was more of an issue and precluded getting usable results for current mission planning. The lack of an ability to optimize the asteroid stay times necessitated using a maximize/minimize radius objective function to improve the NLP solver performance. However, the missing gravity assist feature prohibited getting conservative estimates of an outer approximation as in the single sample return mission.

This thesis showed how using a powerful direct optimization method could quickly reduce the number of targets considered in mission planning for rendezvous type missions, without requiring good guesses of an optimal solution *a priori*. As shown in Table 6, making simple assumptions about inclination and absolute magnitude of possible targets will reduce the number of candidates to be evaluated by several orders of magnitude. A similar process was shown for the two asteroid sample return mission, albeit without the final bounds on a set of possible targets. This methodology was mapped to a different domain using the extremal radius optimization results and can be scalable for higher degrees of fidelity. Thus, this methodology combined with powerful solvers allows anyone with a basic understanding of optimal control theory and

astrodynamics to significantly reduce the mission planning effort required for similar missions. Only more flexible and robust problem formulations are required.

Since this topic was mostly focused on demonstrating methodologies to compute reachable sets any future work should naturally try to increase the fidelity to a 3 DOF example to explore where increased fidelity produces only marginal benefit. Analysis would be easier if DIDO could return the costates and the Hamiltonian for problems that included interior knots.

To make the results of similar methods useful the fidelity must be increased at a minimum to take into account gravity assist effects at Earth and more real NSTAR models that handles the I_{sp} scaling with power [Ref. 17]. The NSTAR model would also benefit from an actual time degradation model vice assuming end-of-life power is the maximum available. Additionally, using an actual launch vehicle performance for trading between C_3 and propellant mass is required. A lesser item that may be useful would be to optimize the drop mass at an Earth flyby for the sample return vehicle. Also, a return declination constraint would be beneficial to ensure any return trajectory could drop off a sample to land at the Utah Test and Training Range. Lastly, developing a method so the optimization scheme could choose to use available gravity assist opportunities from the Earth or moon at any phase of flight might provide interesting results.

APPENDIX A

A. **MATLAB ODE (ORDINARY DIFFERENTIAL EQUATION) SOLVERS**

1. **Solvers for Nonstiff Problems from MATLAB Help File [Ref. 15]**

ODE45 is based on an explicit Runge-Kutta (4,5) formula, it is a one-step solver.

ODE 23 is based on an explicit Runge-Kutta (2,3) pair of Bogacki and Shampine. It also is a one-step solver but may be more efficient than ODE45 at crude tolerances and in the presence of mild stiffness.

ODE 113 is a variable order Adams-Bashforth-Moulton solver. It is a multistep solver (needing several preceding time points to computer current solution) and may be more efficient than ODE45 at stringent tolerances.

2. **Solvers for Stiff Problems from MATLAB Help File [Ref. 15]**

Not all difficult problems are stiff, but all stiff problems are difficult for solvers not specifically designed for them.

ODE15s is a variable-order solver based on the numerical differentiation formulas. Optionally it uses the backward differentiation formulas, BDFs. Like ODE113, ODE15s is a multistep solver.

ODE23s is based on a modified Rosenbrock formula of order 2. Because it is a one-step solver, it may be more efficient than ODE15s at crude tolerances.

ODE23t is an implementation of the trapezoidal rule (TR). This solver is effective if the problem is only moderately stiff and you need a solution without numerical damping.

ODE23tb is an implementation of TR-BDF2, an implicit Runge-Kutta formula with a first stage that is a trapezoidal rule step and a second stage that is a backward differentiation formula of order 2. Like ODE23s, this solver may be more efficient than ODE15s at crude tolerances.

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX B

Typical values for simple Asteroid Rendezvous trajectory to check scaling and balancing to ensure a solution is well conditioned [Ref. 12].

State	low bound	low guess	upper guess	upper bound
r	0.1000	1.0000	1.6470	10.0000
nu	-31.4200	0.0000	14.0100	31.4200
Vr	-10.0000	0.0154	0.0000	10.0000
Vt	-10.0000	1.0250	0.7792	10.0000
Mass	0.1273	1.2730	1.0260	12.7300

State	start value	min value	max value	end value
r	1.0000	1.0000	1.6010	1.6010
nu	0.0000	0.0000	13.9700	13.9700
Vr	0.0089	0.0000	0.0654	0.0000
Vt	1.0280	0.7848	1.0280	0.7903
Mass	1.2730	1.0260	1.2730	1.0260

StateDots	start value	min value	max value	end value
rdot	0.0089	0.0000	0.0654	0.0000
nudot	1.0280	0.4913	1.0280	0.4937
Vrdot	0.0448	-0.0274	0.0563	-0.0016
Vtdot	-0.0065	-0.0579	0.0119	0.0070
Massdot	-0.0148	-0.0148	-0.0075	-0.0075

Control	low bound	low guess	upper guess	upper bound
Thrust	-1.5510	0.0151	0.0067	1.5510
theta	0.0000	4.9300	6.0610	6.2830

Control	start value	min value	max value	end value
Thrust	0.0151	0.0000	0.0151	0.0073
theta	4.9300	0.0000	6.2830	6.0610

Events & Paths	low bound			upper bound
r_0	1.0000			1.0000
nu_0	0.0000			0.0000
C3	0.0000			0.0000
m_0	1.2730			1.2730
Vr_f	0.0000			0.0000
Vt_f	0.0000			0.0000
m_f	1.0260			1.0260
maxThrust	-16860.0000			0.0000
posThrust	0.0000			16860.0000

knots	low bound	low guess	upper guess	upper bound
hard	0.0000	0.0000	21.3100	0.0000
hard	0.0000	0.0000	21.3100	62.8300

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF REFERENCES

1. Melbourne, W. G. and Sauer Jr, C. G., "Optimum Interplanetary Rendezvous With Power-Limited Vehicles," *AIAA Journal*, Vol. 1, No. 1, January 1963.

2. Bryson, Arthur E., Ho, Yu-Chi, *Applied Optimal Control: Optimization, Estimation, and Control*, Hemisphere Publishing, 1975.

3. Prussing, John E., and Conway, Bruce A., Orbital Maneuvers, Oxford University Press, New York, NY, 1993.

4. Stevens, R.E., Design of Optimal Cyclers Using Solar Sails, Master's Thesis, Naval Postgraduate School, Monterey, CA, December 2002.

5. Sauer, Carl (JPL), conversations and email with the author, March - October 2003.

6. Sims, Jon (JPL), conversations with the author, March 2003.

7. Ross, I. M. and Fahroo, F., "User's Manual for DIDO 2003: A MATLAB Application Package for Dynamic Optimization," *NPS Technical Report*, MAE-03-005, Naval Postgraduate School, Monterey, CA, September 2003.

8. Josselyn, S. B., *Optimization of Low Thrust Trajectories With Terminal Aerocapture*, Master's Thesis, Naval Postgraduate School, Monterey, CA, June 2003.

9. Betts, J. T., "Survey of Numerical Methods for Trajectory Optimization," *AIAA Journal of Guidance, Control, and Dynamics*, Vol. 21, No. 2, 1998, pp. 193-207.

10. Stryk, O. and Glocker, M.,"Numerical Mixed-Integer Optimal Control and Motorized Traveling Salesman Problems," *European Journal of Control*, Vol. 35, No. 4, 2001, pp. 519-533.

11. Ross, I. Michael, *AA4850 Class Notes*, Monterey, Spring 2002 and Spring 2003.

12. Ross, I. Michael, conversations with the author, May - November 2003.

13. Kechichian, J.A., "Reformulation of Edelbaum's Low-Thrust Transfer Problem Using Optimal Control Theory," *Journal of Guidance, Control, and Dynamics*, Vol. 20, No. 5, September-October 1997.

14. Vallado, David A., *Fundamental of Astrodynamics and Applications*, Second Edition, Microcosm Press, El Segundo, California, 2001.

15. MATLAB® Help File, "ODE Algorithms", Version 6.1, 18 May 2001.

16. Sutton, G. P. and Biblarz, O., Rocket Propulsion Elements, 7th ed., p. 104, John Wiley & Sons, Inc., 2001.

17. Brophy, John R., and others, "Ion Propulsion System (NSTAR) DS1 Technology Validation Report," NASA Jet Propulsion Laboratory Technical Archives, http://nmp-techval-reports.jpl.nasa.gov/, 28 September 2000.

INITIAL DISTRIBUTION LIST

1.   Defense Technical Information Center
        Ft. Belvoir, VA

2.   Dudley Knox Library
        Naval Postgraduate School
        Monterey, CA

3.   Department Chairman
        Department of Mechanical and Astronautical Engineering
        Naval Postgraduate School
        Monterey, CA

4.   Department of Mechanical and Astronautical Engineering
        ATTN: Professor I. Michael Ross
        Naval Postgraduate School
        Monterey, CA

5.   Ms. Stacy Weinstein
        NASA Jet Propulsion Laboratory
        Pasadena, CA

6.   LT Patrick Croley
        Strategic Systems Programs
        Washington, DC